

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE ELECTRÓNICA



**“LOCALIZACIÓN Y AUTOCALIBRADO
SIMULTÁNEO CON SISTEMAS DE
POSICIONAMIENTO LOCAL ULTRASÓNICOS”**

Autor

Francisco Daniel Ruiz Pereda

Directores

Dr. Jesús Ureña Ureña

Dr. Juan Carlos García García

TESIS DOCTORAL

2011



Universidad
de Alcalá

DEPARTAMENTO DE ELECTRÓNICA
Escuela Politécnica Superior
28805 Alcalá de Henares (Madrid)
Teléfonos: 91 885 65 40 Fax: 91 885 65 91
elden@depeca.uah.es

Dr. Juan Jesús García Domínguez, Director del Departamento de Electrónica de la Universidad de Alcalá,

INFORMA:

Que la Tesis Doctoral titulada "Localización y autocalibrado simultáneo con sistemas de posicionamiento local ultrasónico", presentada por D. Francisco Daniel Ruiz Pereda, y dirigida por el Dr. Jesús Ureña Ureña y el Dr. Juan Carlos García García, cumple con todos los requisitos científicos y metodológicos para ser defendida ante un tribunal.

Alcalá de Henares, a 30 de Mayo de 2011



Fdo.: Juan Jesús García Domínguez.



Dr. Jesús Ureña Ureña, Catedrático de la Universidad de Alcalá, y

Dr. Juan Carlos García García, Profesor Titular de la Universidad de Alcalá

INFORMAN : Que la Tesis Doctoral titulada “Localización y autocalibrado simultáneo con sistemas de posicionamiento local ultrasónico”, presentada por D. Francisco Daniel Ruiz Pereda, y realizada bajo nuestra dirección, dentro del campo de aplicación del **diseño de sistema de autocalibrado y posicionamiento para sistemas ultrasónicos**, reúne los méritos de calidad y originalidad para optar al Grado de Doctor.

Alcalá de Henares, a 30 de Mayo de 2009

Fdo.: Jesús Ureña Ureña



Fdo.: Juan Carlos García García

Resumen

La presente tesis doctoral surge con el objetivo de realizar contribuciones en el calibrado de sistemas de posicionamientos local (LPS) balizados. En la actualidad entre las técnicas más empleadas en la localización en espacios interiores están los sistemas de posicionamientos con balizas, esto es, dentro del área a localizar se instalan una serie de emisores (balizas) y mediante medidas de distancia a los mismos se obtiene la posición de un objeto móvil. Para llevar a cabo este tipo de localización, la gran mayoría de técnicas necesitan conocer la posición de dichas balizas respecto del sistema de referencia que se esté empleando. Calcular estas posiciones suele ser un proceso tedioso y en la mayor parte de los trabajos de localización presentados se asume que es conocida; pero cuando se aborda la implementación real de un sistema de posicionamiento, los errores cometidos en la determinación de la posición de las balizas afectan al error cometido en la posición del móvil tanto como los errores cometidos en la medida de distancias.

La tesis aborda tres grandes aspectos en la temática mencionada. En el primero se propone un nuevo método de posicionamiento para el caso de emplear trilateración hiperbólica. Tras realizar un análisis de las diferentes técnicas de posicionamiento, tanto esféricas como hiperbólicas, se propone un nuevo algoritmo de posicionamiento hiperbólico basado en las propiedades geométricas del operador de Cayley-Menger. Este nuevo algoritmo consigue obtener la posición del nodo móvil con precisiones similares a las de los métodos tradicionalmente utilizados, pero con una menor carga computacional. Además se presenta una versión simplificada del mismo cuando todas las balizas del sistema se encuentran a la misma altura, reduciendo aún más la carga computacional del sistema y eliminando la necesidad de procesos iterativos.

En el segundo bloque de esta tesis se propone un método de calibración para sistemas LPS consistente en tomar varias medidas en diversos puntos dentro del área de trabajo. La principal novedad de este algoritmo es que sólo se necesita conocer la posición real de tres de los puntos de medida mientras que el resto pueden ser desconocidos. Los tres puntos conocidos definen el origen de referencia y eliminan ambigüedades por rotación. Este método permite sólo calibrar sistemas LPS independientes, por eso también se presenta un método de calibración global para varios sistemas LPS independientes en el que mediante un robot móvil se consigue calibrar todos los LPS respecto del mismo sistema

de referencia, fusionando los datos de odometría y la información proporcionada por cada LPS mediante un filtro $H-\infty$.

Finalmente, en el tercer bloque se ha definido un sistema de posicionamiento local ultrasónico, sobre el que se ha podido comprobar experimentalmente la validez de los resultados obtenidos con todos los algoritmos propuestos.

En cumplimiento de la normativa para la concesión de la mención "Doctor Europeo" en el anexo D se encuentra una versión resumida de esta Tesis en lengua inglesa, incluyendo el apartado de conclusiones.

Abstract

The main goal of this PhD thesis is to make contributions to the calibration of a beacon-based local positioning systems (LPS). Nowadays, beacon based localization systems are one of the most common solutions used in indoor localization systems. Inside a coverage area several signal emitters (beacons) are placed in fixed positions and by measuring the distance from the beacons to the mobile node it is possible to calculate its position.

In order to achieve the localization of a mobile node, most of the localization algorithms need to know the beacons' position regarding the reference system that it is being used. Obtaining these positions is often a long and a tedious process, so in most of the papers in the scientific literature the beacons' position are assumed to be known. But when the real implementation of a local positioning system is carried out, the errors in the determination of the beacons' position affects to the error in the mobile node localization in a similar way that the errors made in the measurement of the distances to the beacons themselves.

The PhD thesis can be divided into three major parts. In the first one, a new positioning algorithm for hyperbolic trilateration is proposed. After a detailed analysis of the different localization techniques, both spherical and hyperbolic, a new localization algorithm is proposed that is based on the geometric properties of the Cayley-Menger operator. This algorithm gets the mobile node position with a similar accuracy as the classic methods but with a lower computational load. Also a simplified version of the algorithm when all beacons are at the same height is presented. This version further reduces the computational load of the system and avoids the need of a iterative process.

In the second part, several calibration techniques for local positioning systems are proposed. It is studied the case for a single LPS and the case where a global LPS is composed of various independent LPS. The calibration algorithm calculates the beacons' position taking several measurements from different points of the coverage area. The main contribution of this algorithm is that it only needs to know the actual localization of three measuring points while the rest of the points can be at unknown positions. It is necessary to know the position of three points to define the reference system and to avoid ambiguities

in the solution due to rotations and specularities. This method only allows calibrating a single independent LPS system, so a global calibration system for several independent LPS is also proposed. This method uses a mobile robot to calibrate all the LPS systems regarding the same reference origin by merging the odometry data with the LPS systems information through an $H-\infty$ filter.

Finally, in the last part, it is defined a prototype of an ultrasonic local positioning system. This prototype has been used to verify the results and performance of the different algorithms proposed in this thesis.

In order to get the “European Doctor” mention, in the appendix D appears an extended abstract of this thesis in English, including the conclusion section.

A mi mujer Almudena y mis padres.

Agradecimientos

La realización de esta tesis ha implicado mucho tiempo y trabajo, así como a numerosas personas. Aunque no es posible nombrarlos a todos aquí, deseo al menos mencionar a los que han sido más determinantes para el desarrollo de la tesis.

En primer lugar, quiero dar muy sinceramente las gracias a mis directores de tesis, Jesús Ureña y Juan Carlos García por toda su ayuda, dedicación y su mucha paciencia demostrada en la dirección de esta tesis. Sin ellos realmente no me habría sido posible llegar hasta aquí.

Por otro lado, me gustaría agradecer a Mari Carmen Pérez e Isaac Gude la ayuda prestada en la puesta en marcha del sistema LPS. Sin ellos no hubiera sido posible la realización de las pruebas experimentales.

También me gustaría dar las gracias a la gente de la Universidad UCD de Dublín, especialmente a Chris Bleakley y Juan Ramon Gonzalez por la calida acogida recibida y por toda su ayuda prestada durante la estancia.

Aunque no puedo nombrar a cada uno de ellos, no puedo olvidarme de todos mis compañeros del fondo 31 y 22, no sólo porque ellos han hecho que la tesis sea mucho más llevadera, si no porque han estado dispuestos a ayudarme en todo momento. Me gustaria también dar las gracias a los compañeros del grupo Geintra US&RF por sus consejos, ayuda y la paciencia demostrada al aguantar mis presentaciones.

También quisiera hacer extensivo este agradecimiento a todos los compañeros del Departamento de Electrónica de la UAH, alumnos, profesores y PAS por su ayuda a todos los niveles. A todos ellos, gracias por hacer posible este trabajo.

En el terreno personal, quiero dar las gracias a mis padres, siempre me animaron a estudiar y me ayudaron a afrontar los problemas paso a paso, quiero agradecer todos los consejos que me han dado. También quiero dar las gracias a mis suegros por apoyarme

en todo momento en la realización de esta tesis. Por último quiero dar las gracias a mis amigos, que siguen contando conmigo a pesar de mis ausencias.

Por último mi agradecimiento más especial va para Almudena, por muchos motivos: estar siempre a mi lado; no sólo no quejarse cuando me fuí de estancia, sino venir a visitarme para hacerla más llevadera; y sobre todo animarme y sentirse orgullosa de mí.

A todos ellos, gracias por hacer posible este trabajo.

Índice general

1. Introducción	1
1.1. Entorno de desarrollo de la tesis	2
1.2. Estructura de la tesis	3
2. Estado del Arte y objetivos planteados	5
2.1. Métodos de localización	5
2.1.1. Técnicas para la obtención de la ubicación del objeto móvil . . .	7
2.1.2. Técnicas de medida de distancias	8
2.2. Sistemas de localización en interiores	10
2.2.1. Localización dependiente de la estructura	12
2.2.1.1. Sistemas basados en infrarrojos (IR)	12
2.2.1.2. Sistema basados en ultrasonidos (US)	13
2.2.1.3. Sistemas Ultrawideband (UWB)	15
2.2.1.4. Sistemas basados en marcadores	16
2.2.1.5. Localización RFID	18
2.2.1.6. Localización con redes ZigBee	18
2.2.2. Localización independiente de la estructura	19
2.2.2.1. Localización WiFi	19
2.2.2.2. Localización GSM	21
2.2.2.3. Localización con Bluetooth	21
2.3. Sistemas de Autocalibración para LPS interiores	22
2.3.1. Métodos de obtención de la información	22
2.3.2. Autocalibración empleando posicionamiento inverso	23
2.3.3. Sistemas de Autocalibración basados en métodos iterativos	24
2.4. Objetivos planteados	25
3. Algoritmo de posicionamiento basado en los determinantes de Cayley-Menger	29
3.1. Bideterminantes de Cayley-Menger	29
3.1.1. Propiedades geométricas del operador de Cayley-Menger	30
3.2. Posicionamiento esférico	32
3.3. Posicionamiento hiperbólico	36

3.3.1.	Método 1: Localización con balizas en Z constante (LZC)	38
3.3.2.	Método 2: Localización por tripletas rotadas (LTR)	40
3.3.3.	Método 3: Localización por tripletas con cálculo iterativo (LTI)	44
3.4.	Resultados de simulaciones	46
3.4.1.	Pruebas para los métodos LZC y LTI	47
3.4.2.	Pruebas para los métodos LTR y LTI	52
3.5.	Conclusiones	57
4.	Algoritmo para autocalibración de un sistema LPS balizado	59
4.1.	Introducción	59
4.2.	Algoritmo propuesto para la autocalibración de balizas	61
4.2.1.	Caso esférico	61
4.2.2.	Caso hiperbólico	62
4.2.3.	Diagrama de flujo genérico	64
4.3.	Puntos óptimos de test	65
4.4.	Resultados de simulación	66
4.4.1.	Caso esférico	67
4.4.1.1.	Ubicación de los puntos óptimos	68
4.4.1.2.	Puntos óptimos Vs. Puntos aleatorios	70
4.4.1.3.	Número de puntos de test	76
4.4.2.	Caso hiperbólico	80
4.4.2.1.	Ubicación de los puntos óptimos	81
4.4.2.2.	Puntos óptimos Vs. puntos aleatorios	82
4.4.2.3.	Número de puntos de test	87
4.5.	Conclusiones	91
5.	Calibración global de varios LPS ultrasónicos mediante un robot móvil.	93
5.1.	Calibración para cada LPS individual	93
5.1.1.	Fase de Precalibración	94
5.1.2.	Fase de Calibración	95
5.1.2.1.	Formulación del filtro $H-\infty$	96
5.1.2.2.	Descripción de la odometría del robot	98
5.1.2.3.	Descripción del modelo del LPS	99
5.1.2.4.	Modelado de las fuentes de ruido en el sistema	100
5.1.2.5.	Diagrama de flujo	101
5.1.3.	Fase de Navegación	101
5.2.	Extensión a varios LPS	102
5.2.1.	Mejora de la estimación del punto P_2 con infraestructura adicional	104
5.2.2.	Mejora de la estimación del punto P_2 sin infraestructura adicional	105
5.2.3.	Diagrama de flujo	106

5.3.	Pruebas realizadas bajo simulación	107
5.3.1.	Guiado del robot móvil	107
5.3.2.	Calibración de un sistema LPS individual	111
5.3.3.	Calibración global de varios sistemas LPS	118
5.4.	Conclusiones	122
6.	Pruebas Reales	125
6.1.	Estructura del sistema LPS real	125
6.1.1.	Características de los transductores ultrasónicos empleados	130
6.1.1.1.	Transductor en la etapa de emisión	130
6.1.1.2.	Transductor en la etapa de recepción	131
6.1.2.	Módulo emisor	132
6.2.	Pruebas de posicionamiento con el algoritmo propuesto basado en Cayley-Menger	133
6.2.1.	Pruebas en puntos fijos	134
6.2.2.	Pruebas en movimiento	135
6.3.	Pruebas de autocalibración empleando medidas desde puntos desconocidos	140
6.3.1.	Pruebas para posicionamiento esférico	141
6.3.1.1.	Partiendo de cuatro puntos de test desconocidos	141
6.3.1.2.	Partiendo de ocho puntos de test desconocidos	145
6.3.2.	Pruebas para posicionamiento hiperbólico	148
6.3.2.1.	Partiendo de cuatro puntos de test desconocidos	148
6.3.2.2.	Partiendo de ocho puntos de test desconocidos	150
6.4.	Pruebas de localización y calibración simultáneas empleando el filtro $H-\infty$	153
6.5.	Conclusiones	155
7.	Conclusiones y trabajos futuros	159
7.1.	Conclusiones	159
7.2.	Trabajos futuros	163
7.3.	Publicaciones derivadas de la tesis	164
	Apéndices	167
A.	Método de resolución no lineal de Gauss-Newton aplicado a trilateración	167
A.1.	Trilateración esférica	167
A.2.	Trilateración hiperbólica	169
B.	PDOP	171
B.1.	Cálculo del PDOP de forma teórica	171
B.2.	Cálculo del PDOP de forma experimental	172

C. Función de minimización de MATLAB fmincon	173
D. Extended abstract and conclusions (English version)	175
D.1. Introduction	175
D.1.1. Motivation	175
D.1.2. Background	176
D.1.3. The aims of this thesis	179
D.2. Localization algorithm based on the Cayley-Menger geometric properties	180
D.2.1. Cayley-Menger bideterminant	180
D.2.2. Spherical positioning	180
D.2.3. Hyperbolic trilateration	182
D.2.3.1. Method 1: LZY (Localization at constant Z.	185
D.2.3.2. Method 2: Localization with Rotated Triplets	185
D.2.3.3. Method 3: LTI Localization using an iterative process .	186
D.2.4. Actual test	187
D.3. Calibration algorithm for a beacon based LPS system	189
D.3.1. Spherical version	189
D.3.2. Hyperbolic version	191
D.3.3. Optimal localization of the test points	193
D.3.4. Actual Results	193
D.4. Global calibration of several ultrasonic LPSs using a mobile robot	197
D.4.1. Calibration of a single LPS	197
D.4.1.1. Precalibration	197
D.4.1.2. Calibration	198
D.4.1.3. Navigation	200
D.4.2. Extension to several LPS	200
D.4.3. Actual Results	201
D.5. Conclusions and future works	203
Bibliografía	207

Índice de figuras

2.1. Tipos de posicionamiento. Imagen extraída de [MND06].	5
2.2. Posicionamiento esférico	9
2.3. Trilateración hiperbólica. A es la baliza de referencia, B y C representa el resto de balizas y “ <i>Radiosonde position</i> ” es el punto a localizar. Imagen extraída de http://badc.nerc.ac.uk/data/radiosonde/figures/figfive.html . . .	10
2.4. Estado actual de desarrollo y grado de precisión de las diferentes tecnologías de localización en interiores. Imagen extraída de [HSK04]	11
2.5. Predominio de tecnologías base en posicionamiento en interiores. Imagen extraída de [HSK04]	12
2.6. Active Badget Tags. Imagen extraída de [WHFG92]	13
2.7. Imagen del receptor (izquierda) y de una baliza emisora (derecha) en el sistema Cricket. Imegen extraída de [PMBT01]	14
2.8. Sistema Active Bat. Imagen extraída de: http://www.cl.cam.ac.uk	14
2.9. Sistema Ubisense	16
2.10. Persona que viste un VIS-Tracker (vision-inertial self-tracker) en un entorno plagado de indicadores. Imagen extraída de [NF02]	17
2.11. Vista de la escena (izquierda) y vista de IR (derecha). Imagen extraída de [NKY08].	17
2.12. Resumen del sistema de localización de usuarios empleando indicadores invisibles. Imagen extraída de [NKY08].	17
2.13. Arquitectura de un posible sistema RFID. Imagen extraída de [HVBW01]	18
2.14. Redpin: arquitectura del “ <i>sniffer</i> ” (lector de paquetes) del teléfono móvil. Imagen extraída de [Bol08]	20
2.15. Problema de posicionamiento inverso. (a) Dispositivo de calibración muy grande. (b) Dispositivo de calibración muy alejado. Imagen extraída de [NNHM04]	23
2.16. Método de calibración. Imagen extraída de [NNHM04]	24
3.1. Propiedad geométrica de los determinantes de Cayley-Menger.	30

3.2. Propiedad geométrica de los bideterminantes de Cayley-Menger para secuencias de dos puntos.	31
3.3. Ángulo entre dos rectas	31
3.4. Ángulo definido por dos triángulos	32
3.5. Problema de localización. Obtención de las coordenadas del punto \mathbf{m} a partir de las distancias l_1, l_2, l_3 a sendos puntos $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ de coordenadas conocidas.	33
3.6. Coordenadas Baricéntricas Generalizadas. Obtención de la constante w_j .	34
3.7. Paso de trilateración esférica a hiperbólica	37
3.8. Esquema de posicionamiento para el método LZC.	39
3.9. Esquema de posicionamiento rotando el sistema de referencia (LTR). . . .	41
3.10. Esquema de posicionamiento para el método LTI.	44
3.11. Elección del signo de la derivada.	46
3.12. Distribución de las balizas y los puntos de test para la primera simulación.	47
3.13. Comparativa PDOP obtenido entre los métodos propuestos y la resolución directa por Gauss-Newton, para la primera distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.	48
3.14. Histograma con el porcentaje en diferencia entre los métodos propuestos LZC y LTI y el método de resolución directa para la primera distribución de balizas. Izquierda, método LZC Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton. . .	49
3.15. Distribución de las balizas y los puntos de test para la segunda simulación.	50
3.16. Comparativa PDOP obtenido entre los métodos propuestos y la resolución directa por Gauss-Newton, para la segunda distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.	50
3.17. Histograma con el porcentaje en diferencia entre los métodos propuestos LZC y LTI y el método de resolución directa para la segunda distribución de balizas. Izquierda, método LZC Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton. . .	51
3.18. Distribución de las balizas y los puntos de test para la tercera prueba. . . .	52
3.19. Comparativa PDOP obtenido entre los métodos propuestos LTR (método 2) y LTI (método 3) con el método de la resolución directa por Gauss-Newton, para la tercera distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.	53
3.20. Comparativa PDOP obtenido entre el método LTI y la resolución directa mediante Gauss-Newton, para la tercera distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.	53

3.21. Histograma con el porcentaje en diferencia entre los métodos propuestos LTR y LTI y el método de resolución directa para la tercera distribución de balizas. Izquierda, método LTR Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton.	54
3.22. Distribución de las balizas y los puntos de test para la tercera prueba. . . .	55
3.23. Comparativa PDOP obtenido entre los métodos propuestos y la resolución directa por Gauss-Newton, para la cuarta distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.	56
3.24. Histograma con el porcentaje en diferencia entre los métodos propuestos LTR y LTI y el método de resolución directa para la cuarta distribución de balizas. Izquierda, método LTR Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton.	57
4.1. Esquema de autocalibración. Ejemplo de ubicación de los puntos de test desconocidos y conocidos para realizar la calibración del sistema LPS. . .	60
4.2. Diagrama de flujo de los algoritmos de autocalibración propuestos. . . .	65
4.3. PDOP para ambas distribuciones de balizas en el caso esférico.(a) PDOP para $z=0$ en el área de cobertura cuando se simula la primera distribución de balizas. (b) PDOP para $z=0$ en el área de cobertura cuando se simula la segunda distribución de balizas. Los círculos rojos representan la proyección de las balizas en el plano $z=0$	67
4.4. Representación de la ubicación de las zonas de puntos óptimos para la primera distribución de balizas en el plano $z=0$ para el caso esférico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.	68
4.5. Variación de la ubicación de los puntos óptimos con la altura en la primera distribución de balizas para el caso esférico.	69
4.6. Representación de la ubicación de las zonas de puntos óptimos para la segunda distribución de balizas en el plano $z=0$ para el caso esférico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.	70
4.7. Variación de la ubicación de los puntos óptimos con la altura en la segunda distribución de balizas para el caso esférico.	70
4.8. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\} \text{ cm}$ situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la primera distribución de balizas con posicionamiento esférico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.	72

- 4.9. Comparación de la desviación típica en cada una de las componentes (x, y, z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la primera distribución de balizas con posicionamiento esférico. 73
- 4.10. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\}$ cm situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la segunda distribución de balizas con posicionamiento esférico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas. 75
- 4.11. Comparación de la desviación típica en cada una de las componentes (x, y, z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la segunda distribución de balizas con posicionamiento esférico. 76
- 4.12. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\}$ cm situando los puntos de test desconocidos de forma óptima y empleando la primera distribución de balizas con posicionamiento esférico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima. 77
- 4.13. Comparación de la desviación típica en cada una de las componentes (x, y, z) , aumentando los puntos de medida, para la primera distribución de balizas con posicionamiento esférico. 78
- 4.14. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\}$ cm situando los puntos de test desconocidos de forma óptima y empleando la primera distribución de balizas con posicionamiento esférico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima. 79
- 4.15. Comparación de la desviación típica en cada una de las componentes (x, y, z) , aumentando los puntos de medida, para la segunda distribución de balizas con posicionamiento esférico. 80
- 4.16. PDOP para ambas distribuciones de balizas en el caso hiperbólico.(a) PDOP para $z=0$ en el área de cobertura cuando se simula la primera distribución de balizas. (b) PDOP para $z=0$ en el área de cobertura cuando se simula la segunda distribución de balizas. Los círculos rojos representan la proyección de las balizas en el plano $z=0$ 81
- 4.17. Representación de la ubicación de las zonas de puntos óptimos para la primera distribución de balizas en el plano $z=0$, para el caso hiperbólico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos. 82

4.18. Representación de la ubicación de las zonas de puntos óptimos para la segunda distribución de balizas en el plano $z=0$, para el caso hiperbólico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.	82
4.19. Comparación de la desviación típica en cada una de las componentes (x,y,z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la primera distribución de balizas con posicionamiento hiperbólico.	83
4.20. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\}$ cm situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la primera distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.	84
4.21. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\}$ cm situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la segunda distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.	86
4.22. Comparación de la desviación típica en cada una de las componentes (x,y,z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la segunda distribución de balizas con posicionamiento hiperbólico. .	87
4.23. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\}$ cm situando los puntos de test desconocidos de forma óptima y empleando la primera distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima.	88
4.24. Comparación de la desviación típica en cada una de las componentes (x,y,z) , aumentando los puntos de medida, para la primera distribución de balizas con posicionamiento hiperbólico.	89
4.25. Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\}$ cm situando los puntos de test desconocidos de forma óptima y empleando la segunda distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima.	90
4.26. Comparación de la desviación típica en cada una de las componentes (x,y,z) , aumentando los puntos de medida, para la segunda distribución de balizas con posicionamiento hiperbólico.	91

5.1. Fases propuestas para la calibración de un LPS	94
5.2. Ejemplo de obtención de la posición de una baliza a partir de las distancias medida entre el móvil y la baliza en varios puntos de la trayectoria de precalibración.	95
5.3. Variación de la posición del robot entre dos instantes de tiempo consecu- tivos.	99
5.4. Diagrama de Flujo del método propuesto para la calibración de cada LPS.	101
5.5. Calibración global de varios sistemas LPS	103
5.6. Error en la estimación de las balizas en el segundo LPS debido a los fallos cometidos en la odometría del robot. Los círculos son las proyecciones de las balizas en el plano de medida del robot.. . . .	104
5.7. Empleo de un LPS de apoyo para reducir las zonas donde el robot navega sólo con odometría.	104
5.8. Mejora en la estimación del punto P_2 realizando varias veces el viaje de ida y vuelta entre los LPS con el robot móvil.	106
5.9. Diagrama de flujo para varios LPS.	106
5.10. Recorrido circular realizado en el primer test de navegación.	109
5.11. Ampliación de la parte superior del recorrido realizado en el primer test de navegación.	109
5.12. Recorrido cuadrado realizado en el segundo test de navegación	110
5.13. Ampliación de la esquina superior izquierda del recorrido realizado en el segundo test de navegación.	111
5.14. Recorrido realizado en el área de cobertura en el primer test de calibra- ción. Los círculos son las proyecciones de las balizas en el plano de medida.	112
5.15. Error medio cometido en la posición de las balizas estimadas en el primer test de calibración.	113
5.16. Error cometido en posición y ángulo del robot durante la fase de calibra- ción en el primer test.	113
5.17. Desviación típica en el error en cada una de las componentes X,Y,Z de cada una de las balizas obtenidas después de realizar 100 simulaciones con el recorrido de la primera calibración.	114
5.18. Cálculo de las elipses de error con un intervalo de confianza del 95 % en la posición estimada de las balizas en el primer test. En rojo las elipses obtenidas después de la fase de Precalibración, en verde las obtenidas después de la fase de Calibración.	115
5.19. Recorrido realizado en el área de cobertura en el segundo test de cali- bración. Los círculos son las proyecciones de las balizas en el plano de medida.	115

5.20. Error medio cometido en la posición de las balizas estimadas en el segundo test de calibración.	116
5.21. Error cometido en posición y ángulo del robot durante la fase de calibración en el segundo test.	117
5.22. Desviación típica en el error en cada una de las componentes X,Y,Z de cada una de las balizas obtenidas después de realizar 100 simulaciones con el recorrido de la segunda calibración.	117
5.23. Cálculo de las elipses de error con un intervalo de confianza del 95 % en la posición estimada de las balizas en el segundo test. En rojo las elipses obtenidas después de la fase de Precalibración, en verde las obtenidas después de la fase de Calibración.	118
5.24. Recorrido realizado en la calibración del primer LPS en el primer test de calibración global. Los círculos son las proyecciones de las balizas en el plano de movimiento del robot.	119
5.25. Estimación del punto inicial de calibración del 2º LPS para la primera prueba de calibración global. Los círculos son las proyecciones de las balizas en el plano de movimiento del robot.	120
5.26. Calibración global del 2º LPS tanto si se emplea el método de estimación del punto inicial de calibración como si no se emplea. Los círculos son las proyecciones de las balizas en el plano de medida del robot.	121
5.27. Calibración global del sistema para el segundo test tanto si se emplea el método de estimación del punto inicial de calibración como si no se emplea. Los círculos son las proyecciones de las balizas en el plano de medida del robot.	122
6.1. Esquema del sistema LPS	126
6.2. Esquema funcional del sistema LPS	126
6.3. Estructura de las balizas	127
6.4. Módulo emisor.	128
6.5. Escenario de las pruebas realizadas con el LPS ultrasónico.	129
6.6. Esquema de la zona de trabajo. Además se muestra la proyección de las balizas en el suelo y el origen de coordenadas.	130
6.7. Características del emisor utilizado: patrones de emisión horizontal, vertical y respuesta en frecuencia.	131
6.8. Respuesta en frecuencia del micrófono empleado en la recepción.	131
6.9. Diagrama de bloques del módulo emisor.	132
6.10. Comparativa entre el método esférico basado en los bideterminantes de Cayley-Menger y el método GN esférico para una rejilla de puntos fijos. .	134

6.11. Comparativa entre el método LZC y el método GN hiperbólico para una rejilla de puntos fijos.	135
6.12. Comparativa entre el método LTI y el algoritmo GN hiperbólico para una rejilla de puntos fijos.	135
6.13. Comparativa entre el algoritmo basado en GN esférico y el algoritmo esférico basado en los bideterminantes de Cayley-Menger, realizando un recorrido dentro del área de cobertura.	136
6.14. Diferencia en valor absoluto de las coordenadas X e Y obtenida con el método GN esférico y el algoritmo esférico basado en los bideterminantes de Cayley-Menger.	136
6.15. Comparativa entre el algoritmo basado en Gauss-Newton y el método LZC basado en Cayley-Menger que supone que las balizas están en un plano Z constante, realizando un recorrido dentro del área de cobertura.	137
6.16. Diferencia en valor absoluto de las coordenadas X e Y obtenida con el método LZC (todas las balizas a la misma altura) y el método GN hiperbólico.	138
6.17. Comparativa entre el algoritmo basado en Gauss-Newton y el método LTI basado en Cayley-Menger, realizando un recorrido dentro del área de cobertura.	138
6.18. Diferencia en valor absoluto de las coordenadas X e Y obtenida con el método LTI (método iterativo) y el método GN hiperbólico.	139
6.19. Rejilla para realizar la calibración dentro del área de cobertura.	140
6.20. Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración esférica 3D con 4 puntos de test desconocidos y su posición real.	142
6.21. Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 3D, empleando 4 puntos de test desconocidos y posicionamiento esférico.	143
6.22. Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración esférica 2D con 4 puntos de test desconocidos y su posición real.	144
6.23. Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 2D, empleando 4 puntos de test desconocidos y posicionamiento esférico.	145
6.24. Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración esférica 3D con 8 puntos de test desconocidos y su posición real.	146

6.25. Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 3D, empleando 8 puntos de test desconocidos y posicionamiento esférico.	146
6.26. Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración esférica 2D con 8 puntos de test desconocidos y su posición real.	147
6.27. Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 3D, empleando 8 puntos de test desconocidos y posicionamiento esférico.	148
6.28. PDOP del sistema del posicionamiento local cuando se emplea posicionamiento hiperbólico dentro de la zona de cobertura. (a) PDOP hiperbólico del sistema. (b) HDOP hiperbólico del sistema. (c) VDOP hiperbólico del sistema	149
6.29. Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración hiperbólica 2D con 4 puntos de test desconocidos y su posición real.	150
6.30. Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 2D, empleando 4 puntos de test desconocidos y posicionamiento hiperbólico.	151
6.31. Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración hiperbólica 2D con 8 puntos de test desconocidos y su posición real.	151
6.32. Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 2D, empleando 8 puntos de test desconocidos y posicionamiento hiperbólico.	152
6.33. Estructura de las balizas del LPS empleado en las pruebas reales. Además todas las balizas se encuentran a 3.5 m del suelo.	153
6.34. Recorrido realizado con el robot para realizar la calibración con el método basado en el filtro H- ∞ . Además se emplean las primeras muestras para realizar la precalibración.	154
6.35. Resultados obtenidos después de realizar la autocalibración mediante el método basado en el filtro H- ∞ . (a) Estructura de las balizas después del proceso de precalibración. (b) Resultados obtenidos después de la calibración. La altura real de las balizas es 3.5 m.	154

6.36. Recorrido realizado con el robot para realizar la calibración con el método basado en el filtro $H-\infty$. Se producen dos deslizamientos durante la fase de Calibración. Además se emplean las primeras muestras para realizar la precalibración.	155
6.37. Resultados obtenidos después de realizar la autocalibración mediante el método basado en el filtro $H-\infty$. Se introducen deslizamientos en la odometría del robot en la fase de calibración. (a) Estructura de las balizas después del proceso de precalibración. (b) Resultados obtenidos después de la calibración. La altura real de las balizas es 3.5 m.	156
A.1. Objetivo del método de resolución basado en Gauss-Newton.	167
D.1. Current status of different indoor location technologies: development and accuracy. Image extracted from [HSK04]	177
D.2. Prevalence of based indoor technology. Image extracted from [HSK04] .	177
D.3. Localization problem. Calculate the coordinates of the point ' \mathbf{m} ' using the measured distance l_1, l_2, l_3 to the points ($\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3$) respectively. . . .	181
D.4. From spherical to Hyperbolic trilateration being \mathbf{p}_1 the reference beacon. .	182
D.5. Hyperbolic trilateration localization scheme.	184
D.6. Localization scheme rotating the reference origin.	186
D.7. Localization scheme for the non-linear method.	186
D.8. Ultrasonic LPS.	187
D.9. Comparison between the direct resolution of the equations using Gauss-Newton and the first proposed method based on Cayley-Menger operator, making a path inside the coverage area.	188
D.10. Comparison between the direct resolution of the equations using Gauss-Newton and the third proposed method based on Cayley-Menger operator, making a path inside the coverage area.	188
D.11. Error in centimeters between the beacons coordinates obtained using the 3D spherical calibration algorithm proposed and their real coordinates. .	194
D.12. LPS PDOP when hyperbolic trilateration is used inside the coverage area. (a) Hyperbolic PDOP. (b) Hyperbolic HDOP. (c) Hyperbolic VDOP. . . .	195
D.13. Error in centimeters between the beacons coordinates obtained using the 2D hyperbolic calibration algorithm proposed and their real coordinates. .	196
D.14. Phases of the self-calibration algorithm. Precalibration, Calibration and Navigation.	198
D.15. Example of how obtain the position of one beacon using the distances measured between the mobile robot and the beacon on several points of the trajectory.	198
D.16. Global System calibration of several LPS.	200

D.17. Error in the beacons position estimation in the second LPS due to the noise in the robot odometry. Circles are the projection of the beacons on the ground.	201
D.18. Path performed with the mobile robot in order in order to test the proposed calibration algorithm. Also the points used for the precalibration stage are shown.	202
D.19. Comparision between the beacons obtained in the precalibration and in the calibration stage. The height of the beacons is 3.5 m.	202

Índice de tablas

3.1. Diferencias medias y máximas entre el algoritmo MGN con los algoritmos LZC y LTI en el primer test.	48
3.2. Tiempo necesario para finalizar los test en la primera simulación.	49
3.3. Diferencias medias y máximas entre los algoritmos en el segundo test. . .	51
3.4. Tiempo necesario para finalizar los test en la segunda simulación.	52
3.5. Diferencias medias y máximas entre los algoritmos en el tercer test. . . .	54
3.6. Tiempo necesario para finalizar los test en la tercera simulación.	55
3.7. Diferencias medias y máximas entre los algoritmos en el cuarto test. . . .	56
3.8. Tiempo necesario para finalizar el cuarto test.	57
4.1. Distribución de las balizas para las pruebas de autocalibración bajo simulación	67
4.2. Ubicación de los puntos conocidos para todas las pruebas de autocalibración.	68
4.3. Coordenadas de los puntos óptimos para la primera distribución de balizas en el caso esférico.	68
4.4. Coordenadas de los puntos óptimos para la primera distribución de balizas en el caso esférico.	69
4.5. Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la primera distribución de balizas y empleando un posicionamiento esférico.	71
4.6. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), empleando puntos aleatorios y óptimos con la primera distribución de balizas y posicionamiento esférico.	74
4.7. Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la segunda distribución de balizas y empleando un posicionamiento esférico.	74
4.8. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), empleando puntos aleatorios y óptimos con la segunda distribución de balizas y posicionamiento esférico.	76

4.9. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), variando el número de puntos de test con la primera distribución de balizas y posicionamiento esférico.	78
4.10. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), variando el número de puntos de test con la segunda distribución de balizas y posicionamiento esférico.	80
4.11. Coordenadas de los puntos óptimos para la primera distribución de balizas en el caso hiperbólico.	81
4.12. Coordenadas de los puntos óptimos para la segunda distribución de balizas en el caso hiperbólico.	82
4.13. Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la primera distribución de balizas y empleando un posicionamiento hiperbólico.	83
4.14. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), empleando puntos aleatorios y óptimos con la primera distribución de balizas y posicionamiento hiperbólico.	85
4.15. Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la segunda distribución de balizas y empleando un posicionamiento hiperbólico.	85
4.16. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), empleando puntos aleatorios y óptimos con la segunda distribución de balizas y posicionamiento hiperbólico.	87
4.17. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), variando el número de puntos de test con la primera distribución de balizas y posicionamiento hiperbólico.	89
4.18. Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), variando el número de puntos de test con la segunda distribución de balizas y posicionamiento hiperbólico.	91
5.1. Posición de las balizas para las pruebas de guiado del robot.	108
5.2. Errores medios obtenidos con los diferentes métodos en el primer test de guiado.	110
5.3. Errores medios obtenidos con los diferentes métodos en el segundo test de guiado.	111
5.4. Posición final de las balizas estimadas en el primer test de calibración. . .	113
5.5. Posición final de las balizas estimadas en el segundo test de calibración. .	116
5.6. Posición real de las balizas del segundo LPS.	118
5.7. Error en la posición final de las balizas estimadas el el primer LPS para el primer test de calibración global.	119

5.8. Posición tanto real como estimada del punto inicial de calibración del segundo LPS en la primera prueba de calibración global.	120
5.9. Error en cm de la estimación de la posición de las balizas del segundo LPS en la primera prueba de calibración global.	120
5.10. Localización real de las balizas en ambos LPS en la segunda prueba de calibración global.	121
5.11. Error en cm de la estimación de la posición de las balizas en ambos LPS .	122
6.1. Posición de las balizas en el espacio de trabajo.	133
6.2. Posición de las balizas después de la calibración manual.	141
6.3. Posiciones [x, y, z] de los puntos de test conocidos para las pruebas. . . .	141
6.4. Posiciones [x, y, z] de los 4 puntos de test desconocidos para la prueba de autocalibración esférica.	141
6.5. Posición de las balizas después de la autocalibración esférica 3D con 4 puntos de test desconocidos.	142
6.6. Posición de las balizas después de la autocalibración esférica 2D con 4 puntos de test desconocidos.	144
6.7. Posición de los 8 puntos de test desconocidos para la realización de las pruebas de calibración esférica.	145
6.8. Posición de las balizas después de la autocalibración esférica 3D con 8 puntos de test desconocidos.	145
6.9. Posición de las balizas después de la autocalibración esférica 3D con 8 puntos de test desconocidos.	147
6.10. Posiciones [x, y, z] de los 4 puntos de test desconocidos para la prueba de autocalibración hiperbólica.	149
6.11. Posición de las balizas después de la autocalibración hiperbólica 2D con 4 puntos de test desconocidos.	149
6.12. Posición de los 8 puntos de test desconocidos para la realización de las pruebas de calibración hiperbólica.	151
6.13. Posición de las balizas después de la autocalibración hiperbólica 2D con 8 puntos de test desconocidos.	152
D.1. Beacons localization after the manual calibration	193
D.2. Coordinates [x, y, z] of the known test points.	193
D.3. Position of the 8 unknown test points for the spherical calibration test, all units are in meters.	194
D.4. Beacons localization after the 3D spherical calibration with 8 unknown test points.	194
D.5. Position of the 8 unknown test points for the hyperbolic calibration test, all units are in meters.	196

D.6. Beacons localization after the 2D hyperbolic calibration with 8 unknown test points. The coordinate Z is known.	196
---	-----

Lista de símbolos

$\ \mathbf{v}\ $	Módulo del vector \mathbf{v} .
$\triangle D_k$	Velocidad lineal del robot en el instante k .
$\triangle \theta_k$	Incremento en el ángulo del robot en el instante k .
$\triangle d(\mathbf{p}, \mathbf{q})$	La diferencia de distancia existente entre un punto (a posicionar) y otros dos puntos: el punto \mathbf{p} y el punto \mathbf{q} .
\times	Producto vectorial.
σ	Desviación típica.
σ^2	Varianza.
θ_k	Ángulo del robot en el instante k .
γ	Parámetro de ajuste del filtro $H-\infty$.
$ \mathbf{A} $	Determinante de la matriz \mathbf{A} .
\mathbf{A}_k	Modelo dinámico del sistema de posicionamiento.
$A(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$	Área con signo del triángulo formado por los puntos \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 .
\mathbf{C}_k	Modelo dinámico de la odometría del robot.

$D()$	Operador del bideterminante de Cayley-Menger.
$E[\mathbf{A}]$	Esperanza matemática de la matriz \mathbf{A} .
F	Función de error.
\mathbf{I}	Matriz identidad.
\mathbf{K}	Ganancia del filtro $H-\infty$.
\mathbf{P}	Matriz de covarianza del filtro $H-\infty$.
\mathbf{P}_2	Punto donde comienza la calibración global del segundo LPS.
$\text{Trace}(\mathbf{A})$	Traza de la matriz \mathbf{A}
\mathbf{V}	Matriz de ponderación del ruido de las medidas de distancia en el filtro $H-\infty$.
\mathbf{W}	Matriz de ponderación del ruido en la odometría en el filtro $H-\infty$.
\mathbf{b}_i	Posición de la baliza i .
$\text{cov}(\mathbf{A})$	Matriz de covarianza de la matriz \mathbf{A} .
$d(\mathbf{p}, \mathbf{q})$	Distancia euclídea entre el punto \mathbf{p} y el punto \mathbf{q} .
$\mathbf{g}^{(i)}$	Posición del objeto obtenida con la tripleta i .
$\mathbf{g}^{(iR)}$	Posición del objeto obtenida con la tripleta i una vez se ha rotado el sistema de referencia.
l	Distancia desde el objeto a localizar a la baliza de referencia.
\mathbf{pai}	Vector que contiene los valores de las variables estimadas en la calibración mediante Gauss-Newton.

$\hat{\mathbf{q}}$	Vector de estado en el filtro $H-\infty$.
r_{im}	Distancia euclídea entre el punto i y el punto m .
\hat{r}_{im}	Distancia euclídea entre el punto estimado i y el punto estimado m .
w_i	Constante de ponderación en el cálculo de las coordenadas baricéntricas.

Lista de abreviaturas

AOA	<i>Angle of Arrival</i> (Ángulo de llegada).
API	<i>Application Programming Interface</i> (Interfaz de programación de aplicaciones).
ASIC	<i>Application Specific Integrated Circuits</i> (Circuitos de aplicación específica).
BPSK	<i>Binary Phase Shift Keying</i> (Modulación binaria por desplazamiento de fase).
CCD	<i>Charge Coupled Device</i> (Dispositivo de carga acoplada).
CDMA	<i>Code Division Multiple Access</i> (Acceso múltiple por división en el código).
DSP	<i>Digital Signal Processor</i> (Procesador de señales digitales).
FPGA	<i>Field Programmable Gate Array</i> (Matriz de puertas programable).
GID	<i>Globally unique IDentifiers</i> (identificador único global).
GN	<i>Gauss-Newton</i> .
GPS	<i>Global Positioning System</i> (Sistema de posicionamiento global).
GSM	<i>Global System for Mobile communications</i> (Sistema global para las comunicaciones móviles).
ILS	<i>Indoor Localization System</i> (Sistema de localización en interiores).
IR	<i>InfraRed</i> (Infrarojo).
LAN	<i>Local Area Network</i> (Red de área local).

LOS	<i>Line Of Sight</i> (Línea de visión).
LPS	<i>Local Positioning System</i> (Sistema de posicionamiento local).
LQ	<i>Link Quality</i> (Calidad del enlace).
LTl	<i>Localización por Tripletas mediante minimización Iterativa.</i>
LTR	<i>Localización por Tripletas Rotadas.</i>
LZC	<i>Localización para plano Z Constante.</i>
MAC	<i>Media Access Control</i> (Control de acceso al medio).
MGN	<i>Método de resolución de las ecuaciones de distancia por Gauss-Newton.</i>
PDA	<i>Personal Digital Assistant</i> (Asistente digital personal).
PDOP	<i>Position Dilution of Precision</i> (Dilución de la precisión en la posición).
RF	<i>Radio Frequency</i> (Radiofrecuencia).
RFID	<i>Radio Frequency IDentification</i> (Identificación por radiofrecuencia).
RSS	<i>Received Signal Strength</i> (Fuerza de la señal recibida).
RSSI	<i>Received Signal Strength Indication</i> (Indicación de fuerza de señal de recepción).
SNR	<i>Signal to Noise Ratio</i> (Relación señal ruido).
TDOA	<i>Time Difference of Arrival</i> (Diferencias de tiempo de vuelo).
TOA	<i>Time Of Arrival</i> (Tiempo de llegada).
TOF	<i>Time Of Flight</i> (Tiempo de vuelo).
UWB	<i>UltraWideBand.</i>
WSN	<i>Wireless Sensor Networks</i> (Redes de sensores inalámbricos).

Capítulo 1

Introducción

Los sistemas de localización son aquellos que tratan de obtener la posición de un objeto o de una persona en algún instante de tiempo. Estos sistemas están ganando importancia debido a la gran cantidad de aplicaciones que ofrecen. Por ejemplo los sistemas de localización podrían ser útiles en las siguientes situaciones: avisar al personal médico cuando un paciente se encuentre en una situación de emergencia (en hospitales o para personas mayores); permitir encontrar a un niño perdido en un parque temático o centro comercial; poder conocer la localización de un paquete enviado en todo momento y no sólo cuando llega a ciertos checkpoints; etc.

Debido a todas estas consideraciones, grandes empresas como Mitsubishi, Cisco, IBM, Microsoft, Motorola... están invirtiendo en tecnologías de localización. De hecho, según un estudio de mercado de IDTechex [IDT09] en el año 2009 las empresas que ofrecen servicios de posicionamiento se han triplicado y se estima que el mercado crezca desde los 153 Millones de dólares en 2009 hasta los 2.6 Billones de dólares en 2018.

Alguna de las tecnologías empleadas en los sistemas de localización son:

- Infrarrojos
- Ultrasonidos
- Radiofrecuencia, en sus diversas variantes, por ejemplo:
 - WiFi
 - UltraWideBand (UWB)
 - ZigBee
- Sistemas por satélite, GPS.
- Sistema basados en telefonía móvil GSM.

La mayoría de estos sistemas requieren el empleo de balizas situadas en el entorno. Por ejemplo en Wi-Fi se emplean los puntos de acceso ubicados en un edificio; en GSM se

emplean las estaciones base ubicadas en el entorno; con Ultrasonidos e Infrarojos se deben situar una serie de balizas en el entorno donde se quiera efectuar la localización.

En la mayor parte de estos sistemas para poder efectuar la localización es necesario conocer la posición real de las balizas y ésto se suele hacer midiendo de forma manual su ubicación respecto al sistema de referencia. La necesidad de este proceso de medidas hace que estos sistemas sean poco portables y adaptables.

En esta tesis se analizan los distintos algoritmos existentes en la literatura y se propone un algoritmo de posicionamiento para el caso de emplear trilateración hiperbólica que ofrece la misma precisión que los métodos tradicionales pero con una menor carga computacional.

Además se aborda el problema de la calibración de un sistema de localización, esto es, obtener la posición de las balizas para después poder llevar a cabo la localización de objetos o personas. Se propondrá un método para poder calibrar el sistema de una forma más eficiente que la medida manual de la posición de las balizas. También también se aborda el problema de la calibración global de sistemas de posicionamiento en entornos extensos, en los cuales, dentro de un mismo entorno, existen varios sistemas independientes que no comparten su área de cobertura y se desea que todos estén referenciados respecto al mismo sistema de coordenadas.

Cabe destacar que, aunque los distintos métodos propuestos han sido desarrollados para mejorar las prestaciones de un sistema sensorial ultrasónico, su uso puede extenderse a cualquier tecnología que calcule la posición del objeto mediante tiempos de vuelo (TOA) o diferencias de tiempos de vuelo (DTOA).

1.1. Entorno de desarrollo de la tesis

Este trabajo se ha llevado a cabo en el Departamento de Electrónica de la Universidad de Alcalá, que acumula una amplia experiencia en el diseño de sistemas sensoriales aplicados a la Robótica Móvil. En concreto, la tesis se ha valido de la experiencia adquirida en varios proyectos de investigación desarrollados en dicho departamento:

- **PARMEI** (Posicionamiento absoluto de robots móviles en espacios interiores; referencia DIP2003-08715-C02-01), este proyecto se desarrolló conjuntamente con el instituto de Automática Industrial del Consejo Superior de Investigaciones Científicas (CSIC) durante los años 2004-2006. El objetivo principal de este proyecto fue la elaboración de nuevos métodos y técnicas para determinar la posición absoluta de robots móviles en espacios interiores; para ello se utilizaron sensores de ultrasonidos e infrarrojos distribuidos en el entorno de movimiento del robot. Se consiguió separar la información proveniente de cada emisor mediante CDMA, así como trabajar en condiciones de baja potencia emitida y eliminar los problemas del multicamino, propio del empleo de señales ultrasónicas en espacios interiores.

- **RESELA** (Integración de redes de sensores acústicos, de visión y RFID para la localización en ambientes inteligentes; referencia: TIN2006-14986-CO2-01) se trató de un proyecto multidisciplinar también en colaboración con el Instituto de Automática Industrial del CSIC durante los años 2006-2009. Los esfuerzos de este proyecto se orientaron a aportar funciones de detección de presencia, localización física e interacción entre usuarios; para ello se contó con tres tecnologías sensoriales complementarias: acústica, cámaras de visión y RFID. Se abordaron los aspectos sensoriales de procesamiento de señal y algoritmos avanzados de alto nivel, así como aspectos de escalabilidad, coste de integración, mantenimiento, comunicación de datos y flexibilidad de las redes sensoriales diseñadas.
- **LEMUR** (Localización continua en entornos extensos mediante ultrasonidos y radiofrecuencia; referencia: TIN2009-14114-C04-01) se trata de un proyecto en el que también colaboran el Centro de Automática y Robótica, la Universidad de Valladolid y la universidad de Extremadura. El proyecto comenzó en el año 2009 y su finalización está prevista para el año 2012. Su objetivo es conseguir avances fundamentales en el desarrollo de sistemas de posicionamiento local (LPS) que operen de modo transparente al usuario en espacios extensos (incluyendo entornos interiores y exteriores restringidos), mediante la combinación de tecnologías ultrasónicas y de radiofrecuencia, y permitiendo una localización continua y robusta mediante la conmutación o fusión dinámica entre plataformas.

Finalmente, desde el punto de vista económico, esta tesis ha sido posible gracias a la financiación del Ministerio de Educación y Ciencia a través de una beca de formación de personal investigador (FPI) adjudicada en la convocatoria de 2007. Además estas ayudas permitieron la realización de una estancia de 3 meses que tuvo lugar en 2009 en la UCD (University College Dublin) School of Computer Science & Informatics en Dublín, Irlanda.

1.2. Estructura de la tesis

Esta tesis se ha dividido en siete capítulos, atendiendo a los diferentes temas que han sido abordados en su desarrollo.

Después de esta introducción, en el capítulo 2 se lleva a cabo un análisis de los diferentes métodos que existen actualmente para llevar a cabo la localización de un objeto móvil. A continuación se procede a analizar de forma más profunda los métodos de localización basados en medidas de distancia. Por otro lado se revisan los sistemas de detección en interiores, primero se analizan los sistemas dependientes de la estructura para a continuación analizarse los sistemas independientes de la estructura. Una vez analizados dichos sistemas se observa que muchos de ellos suponen el conocimiento de la posición de los

emisores del sistema y se analizan las diferentes técnicas que existen para tratar de encontrar las posiciones de los emisores. Finalmente, teniendo en cuenta las limitaciones de los sistemas de calibración actuales, se ha procedido a establecer los objetivos de esta tesis.

En el capítulo 3 se lleva a cabo una propuesta de posicionamiento hiperbólico partiendo del método para posicionamiento esférico propuesto por [TR05]. Este método presenta la ventaja de que es capaz de obtener la localización del nodo móvil de forma directa y con la misma precisión que los métodos iterativos ampliamente utilizados [YTBJ08]. Además se realizan diversas pruebas para demostrar las propiedades de los algoritmos propuestos.

El capítulo 4 presenta la propuesta de un algoritmo de calibración para un sistema de posicionamiento local (LPS). Se proponen dos variantes del algoritmo: una para cuando se empleen técnicas de trilateración esférica; y otra para cuando se empleen técnicas de trilateración hiperbólica. Estos algoritmos calculan la posición de las balizas tomando medidas desde diversos puntos de test dentro del área de cobertura del sistema de posicionamiento local. De todos estos puntos el algoritmo sólo necesita conocer la posición de tres de ellos para evitar ambigüedades en la solución, mientras que la posición del resto puede permanecer desconocida para el mismo. Además se propone un método para, a partir de la geometría del sistema, poder obtener zonas óptimas para situar los puntos de test desconocidos. Al final del capítulo se realizan una serie de pruebas para evaluar cómo afecta al algoritmo el emplear las zonas óptimas y el número de puntos de test empleados para la calibración.

En el capítulo 5 se aborda un nuevo método de calibración global para varios LPS independientes, esto es, varios sistemas independientes calibrados respecto un mismo sistema de referencia global. El método propuesto emplea un robot móvil para fusionar los datos de la odometría del robot con la información del LPS mediante un filtro $H-\infty$ y de esa forma poder obtener la posición de las balizas respecto al mismo sistema de referencia. Además se propone una metodología para minimizar los errores en las zonas donde el robot móvil se mueve sin apoyo de ningún LPS. Por último el algoritmo propuesto también es capaz de guiar al robot móvil una vez se haya realizado la calibración de todos los sistemas.

En el capítulo 6 se presentan las pruebas obtenidas en un LPS real. Dicho LPS está compuesto por cinco balizas ultrasónicas, cada una de ellas emitiendo un código binario único modulado en BPSK. Este LPS se emplea para verificar las propiedades de cada uno de los métodos propuestos en los capítulos anteriores.

Por último en el capítulo 7 se resumen las conclusiones y aportaciones realizadas en esta tesis. Asimismo, se describen posibles líneas futuras de investigación que pueden derivarse del trabajo realizado. Se incluye además un listado de las publicaciones extraídas del trabajo de investigación llevado a cabo.

Estado del Arte y objetivos planteados

En el capítulo anterior ya ha sido introducido el objetivo principal de esta tesis, consistente en la propuesta de localización y autocalibrado de un sistema de posicionamiento local basado en ultrasonidos. Antes de avanzar en el estudio de este sistema, es necesario analizar el estado del arte en localización de objetos móviles y sistemas de autocalibración para sistemas de posicionamiento local (LPS).

2.1. Métodos de localización

El objetivo principal de un sistema de posicionamiento es obtener la posición de un objeto móvil dentro de un sistema de coordenadas dado. Estos sistemas se pueden dividir básicamente en función de la técnica empleada para determinar la posición de un objeto [MND06], según puede observarse en la figura 2.1:

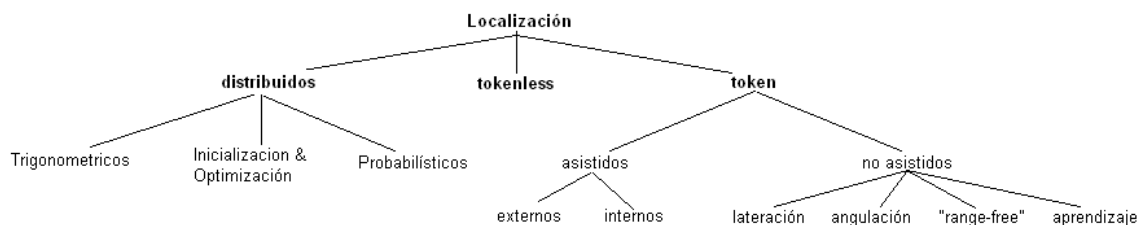


Figura 2.1: Tipos de posicionamiento. Imagen extraída de [MND06].

- **Sistemas tokenless.** En esta aproximación el objeto no porta ningún dispositivo adicional para ayudar en la localización. Toda la información debe ser inferida de las características naturales del objeto mediante los dispositivos del LPS instalados en el entorno. No requieren la colaboración del objeto móvil para obtener su posición [ADB⁺04, OA00]
- **Sistemas token.** En esta aproximación el objeto móvil debe llevar algún dispositivo para poder obtener su localización (RFID, Ultrasonidos, PDA...), por lo que es necesaria su cooperación para llevar a cabo la localización. Estos sistemas a su vez pueden dividirse en:

- Sistemas no asistidos. No hay instalado ningún elemento en el entorno para llevar a cabo la localización, el objeto móvil debe recoger toda la información por sí mismo e inferir su posición a partir de ella. La información necesaria la puede obtener del exterior [HHS⁺02, YB96] o de lecturas de sus sistemas internos [FAM⁺05], como es el caso de los sistemas dead-reckoning.
- Sistemas asistidos. Existe algún dispositivo instalado en el entorno para obtener la posición del objeto móvil. Dentro de estos sistemas se pueden encontrar diversas implementaciones: sistemas basados en medidas de distancias [BP03, PMBT01]; basados en medidas de ángulos [NN03]; sistemas “range-free” que posicionan el objeto por su proximidad a un nodo del sistema [BW04]; por último existen sistemas que emplean técnicas de aprendizaje en puntos específicos del área de cobertura para luego poder extrapolar esos datos y obtener la posición del objeto móvil [LW07, Sim03].
- **Sistemas distribuidos.** En estos sistemas un gran número de sensores son ubicados en el entorno y la posición de cada uno de ellos no es conocida. Los nodos calculan su posición relativa respecto a sus nodos vecinos y esta información se distribuye por la red sensorial. Existen principalmente tres métodos para solucionar este problema.
 - Métodos trigonométricos. Estos sistemas emplean propiedades trigonométricas para obtener la posición de los nodos [SRB01]. Este tipo de soluciones presentan los siguientes inconvenientes:
 1. Los errores en la localización de los nodos se van acumulando cada vez que se añade un nuevo nodo a la lista.
 2. Muchas distribuciones de sensores pueden tener varias soluciones trigonométricas posibles, por lo que la solución encontrada puede no coincidir con la distribución real.
 - Métodos basados en Iniciación y Optimización. Estos algoritmos [GK05, Ma06] se desarrollan en dos fases. En la primera fase se obtiene la topología general formada por la red de sensores. En la segunda fase las localizaciones obtenidas en la primera fase son iterativamente ajustadas para minimizar el error global de las distancias medidas.
 - Métodos basados en probabilidad. Estas técnicas suponen que no se puede conocer la posición exacta de los nodos de la red y emplean funciones de densidad de probabilidad [MMKP02]

Además de todo lo anterior los sistemas de localización se pueden clasificar según su privacidad:

- **Sistemas Centralizados.** La localización se efectúa en la infraestructura de localización. Por ejemplo la operadora de móviles emplea las estaciones base para obtener la posición de un usuario.
- **Sistemas Privados.** La localización se realiza en el dispositivo del usuario. Un ejemplo de este tipo de sistemas es el GPS.

2.1.1. Técnicas para la obtención de la ubicación del objeto móvil

En [SJP⁺09] se clasifican los sistemas LPS en 4 categorías según la técnica matemática que se emplea para la obtención de la posición del objeto a localizar:

- **Métodos geométricos.** Estos métodos se pueden emplear cuando se pueden medir las distancias, diferencias de distancias o los ángulos entre las balizas y el objeto móvil. Por ejemplo la distancia absoluta entre la baliza i de coordenadas \mathbf{x}_i , y el objeto móvil de coordenadas \mathbf{x} sería $r_i = \|\mathbf{x} - \mathbf{x}_i\| + e_i$, donde e_i representa el error en las medidas. Incluso en esta situación tan simple el sistema de ecuaciones que se puede plantear es no lineal y de forma general sobredimensionado por lo que no es posible resolverlo de una forma cerrada.

En [NMH98], [Caf00] y [NG97] se presentan diversos métodos que linealizan el sistema de ecuaciones para tratar de encontrar una solución cerrada. El problema de estos métodos es que en el proceso de linealización se pierde información y no ofrecen una solución óptima, de hecho las soluciones suelen presentar un sesgo aunque el error de medida no lo tenga.

Otra forma de solucionar el problema es mediante estimadores de máxima verosimilitud. En [Foy76] se emplea una descomposición en series de Taylor para encontrar la posición del objeto móvil.

- **Minimización de una función de coste.** En general estos métodos consisten en encontrar el mínimo de una función de coste asociada a la localización del objeto. Normalmente se emplean los métodos de minimización de Gauss-Newton o Levenberg-Marquardt. Algunos ejemplos de estos métodos se pueden encontrar en [CS98, MLG⁺08, NPJ08]. El principal inconveniente de este método es que la convergencia en la solución depende fuertemente de la inicialización del algoritmo.
- **Métodos *Fingerprinting*.** Estos métodos de resolución se emplean mayoritariamente en sistemas LPS de radiofrecuencia; la información de entrada al algoritmo de localización es la amplitud de la señal de las balizas recibida en el nodo móvil (RSS).

Constan normalmente de dos fases: una etapa previa de calibración, donde se toman medidas a través de una rejilla de posiciones dentro del área de cobertura y se almacena la potencia de la señal recibida de las estaciones base; una vez realizada la calibración se pasa a la fase de localización propiamente dicha. En esta última fase el objeto móvil mide las potencias de la señal de las estaciones base y las compara con la base de datos almacenada en la fase de calibración; aquella posición de las almacenadas cuyas lecturas de RSS se correspondan mejor con las lecturas actuales se toma como la posición más probable del móvil. Normalmente para calcular esta posición se emplea métodos de aprendizaje estadístico [FL08, Yim08, BB05]

- **Métodos Bayesianos.** En estos métodos la posición del objeto móvil en el instante t se modela como una función de probabilidad basada en las medidas obtenidas hasta dicho instante. La estimación de la posición del objeto móvil se lleva a cabo en dos fases:

- Etapa de predicción: en esta fase se estima una primera posición del objeto móvil con los datos anteriores y el modelo de movimiento del objeto.
- Etapa de corrección: en esta etapa se corrige la posición del objeto móvil con las medidas obtenidas.

Existen muchas implementaciones de estos métodos, siendo el filtro de Kalman el método más sencillo, pero una de las implementaciones más eficientes de estos métodos son los Filtros de Partículas [KSJ⁺07, MSG⁺05].

2.1.2. Técnicas de medida de distancias

En la sección anterior varias de las técnicas más usadas para obtener la ubicación del objeto móvil hacen uso de información de medida de distancias a una serie de balizas de referencia. Hay varias alternativas para medir la distancia de un objeto a un punto de referencia:

- **Por medida de tiempo de vuelo (TOF o TOA).** En este tipo de posicionamiento se calculan las distancias absolutas desde el objeto móvil hasta los puntos de referencia. Para conocer las distancias a las balizas se necesita conocer la velocidad de propagación en el medio y el instante de inicio de emisión de cada una de ellas. Esto hace necesario el disponer de algún mecanismo de sincronización que permita determinar esta situación. Por ejemplo en [WJH97] se emplea un emisor RF para avisar que las balizas ultrasónicas comienzan a emitir.

Una vez obtenidas las distancias absolutas entre balizas y móvil, la posición del mismo se puede calcular como la intersección de las circunferencias centradas en las posiciones de las balizas y cuyo radio es la distancia medida.

En la figura 2.2 se muestra un ejemplo, en el que se dispone de tres balizas (**P1**, **P2**, **P3**) siendo las distancias medidas (r_1 , r_2 , r_3). La intersección de las dos primeras circunferencias da dos posibles ubicaciones (A y B), mientras que la intersección con la tercera circunferencia sitúa al móvil en la posición B.

La principal ventaja de este método de posicionamiento es su elevada precisión, pero tiene el inconveniente mencionado anteriormente de necesitar de un mecanismo de sincronización. Otro problema relacionado con la sincronización es el derivado de la introducción de retardos en los transductores de ultrasonidos: desde que el sistema señaliza que empieza a emitir hasta que las balizas realmente comienzan la emisión de los ultrasonidos puede existir un retardo que debe ser caracterizado para evitar errores en el posicionamiento [DBS96].

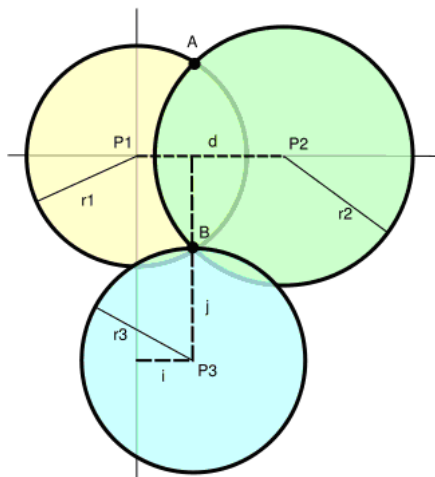


Figura 2.2: Posicionamiento esférico

- **Por medida de diferencias de tiempos de vuelo (TDOF).** Este tipo de técnicas obtienen la posición del móvil conociendo el incremento de distancias que existe desde la recepción de la señal de una baliza tomada como referencia. A partir de estas medidas se pueden trazar diferentes hipérbolas cuyos focos son la baliza con la que se ha calculado la medida y la baliza de referencia. La posición del móvil se encontrará en la intersección de dichas hipérbolas (Figura 2.3). La mayor ventaja de este método consiste en que ya no es necesaria una sincronización entre el objeto móvil y las balizas, pero a cambio, para realizar el posicionamiento se necesita como mínimo una baliza adicional. Otro inconveniente de este método es que la precisión obtenida es generalmente menor que en el caso esférico [DBS96].

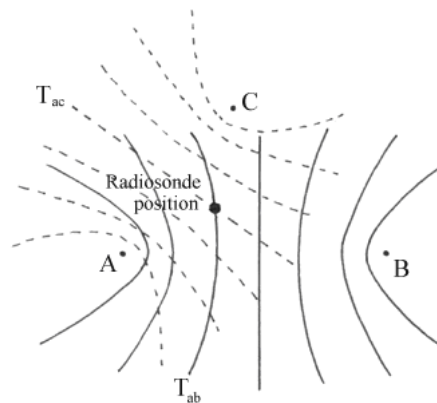


Figura 2.3: Trilateración hiperbólica. A es la baliza de referencia, B y C representa el resto de balizas y “Radiosonde position” es el punto a localizar. Imagen extraída de <http://badc.nerc.ac.uk/data/radiosonde/figures/figfive.html>

- **Procesamiento de Imágenes.** Estas técnicas se emplean cuando los dispositivos instalados en el entorno son cámaras. Existen numerosas alternativas; por ejemplo una posibilidad es usar la medida del grado de enfoque de la imagen: se mueve una lente situada delante de un sensor CCD (Charge Coupled Device) hasta enfocar la imagen, la distancia entre el sensor CCD y la lente está relacionada con la distancia entre el objeto móvil y la cámara [LMP⁺09].
- **Mediante medidas de la amplitud de la señal recibida.** En este tipo de técnicas cada baliza emite una señal de características conocidas. El receptor, al capturar esta señal, obtiene su amplitud y calcula la distancia al punto de referencia usando un modelo de la atenuación de la señal con la distancia. Este método requiere una caracterización previa del medio en el que se transmite la señal, ya sea estudiando la atenuación en el mismo o realizando una calibración del entorno.

2.2. Sistemas de localización en interiores

El creciente interés por sistemas y servicios sensibles a la localización ha hecho que proliferen la investigación en tecnologías de localización de precisión. La localización en interiores es un área de investigación activa dentro de la comunidad científica; asimismo están empezando a surgir una gran variedad de sistemas comerciales que se aprovechan del conocimiento de la localización del usuario. Diversas características permiten diferenciar las soluciones existentes; un ejemplo podría ser la tecnología empleada (RF, IR, visión...), los requerimientos de línea de visión (LOS), la precisión o la escalabilidad. Algunos ejemplos de aplicaciones en los que se requiere una localización precisa en interiores son:

- Servicios sensibles a la localización por ejemplo: publicidad, entretenimiento, etc.

- Detección de pacientes en hospitales.
- Localización de interiores para socorro o emergencias.
- Conocimiento de la localización del stock dentro de grandes almacenes.

La información de posición es muy significativa y valiosa para muchas aplicaciones dentro del campo de la computación ubicua, es un elemento muy importante para determinar el contexto o la actividad que realiza el usuario. Mientras que determinar la posición de un dispositivo móvil en exteriores es posible en la mayoría de las situaciones con GPS (Global Positioning System), no hay un sistema igualmente fiable, preciso y fácil de usar para entornos interiores. El GPS funciona bien en exteriores pero no es aplicable en interiores o en áreas con visión limitada del cielo.

Para poder abordar la problemática anterior se han desarrollado diversos sistemas empleando una gran variedad de tecnologías. En esta sección se han clasificado las técnicas de localización en interiores en dos grandes grupos: los que utilizan infraestructura dedicada para la localización, que debe ser instalada en el edificio, y los que se basan en infraestructuras de comunicaciones ya existentes. En la figura 2.4 se muestra el grado de precisión de cada una de las tecnologías y su estado de desarrollo actual. Además en la figura 2.5 se muestra el predominio de cada una de las tecnologías dentro de los sistemas de posicionamiento.

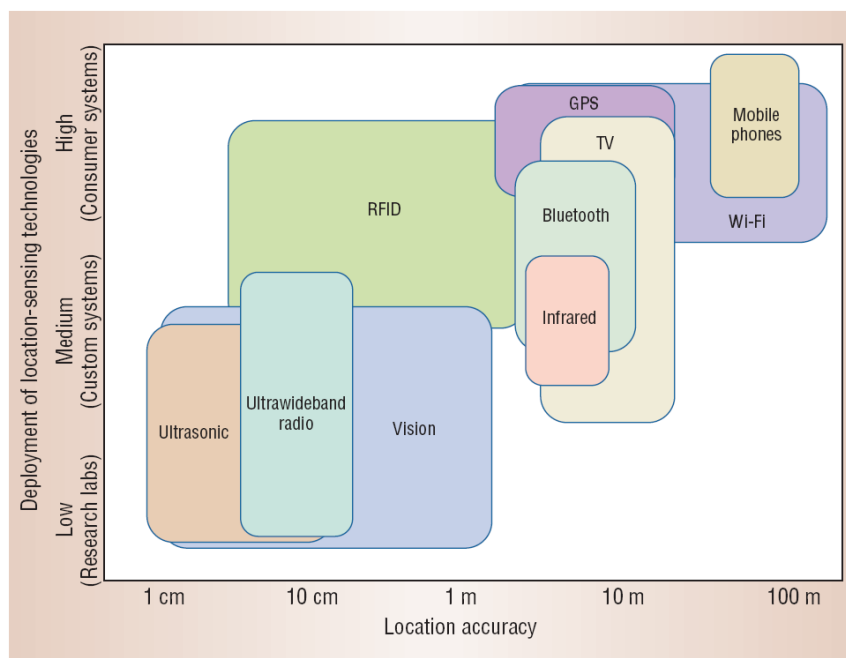


Figura 2.4: Estado actual de desarrollo y grado de precisión de las diferentes tecnologías de localización en interiores. Imagen extraída de [HSK04]

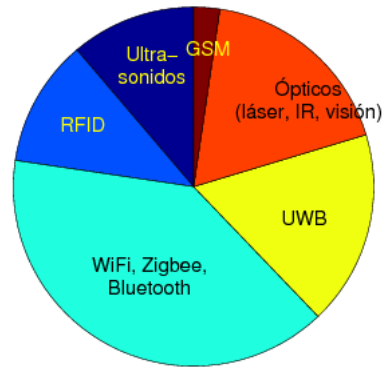


Figura 2.5: Predominio de tecnologías base en posicionamiento en interiores. Imagen extraída de [HSK04]

2.2.1. Localización dependiente de la estructura

La mayor desventaja de estas técnicas de localización es que requieren una infraestructura específica que cubra aquellas áreas en las que se vayan a necesitar los servicios de localización. Existen numerosas soluciones para localización en interiores que requieren de nueva infraestructura entre las que se encuentran [PUH⁺07], [Che08] y [Kar06]. En la mayoría de los casos el coste y esfuerzos de instalación son la mayor desventaja para despliegues a gran escala, particularmente en entornos domésticos.

2.2.1.1. Sistemas basados en infrarrojos (IR)

Active Badge [WHFG92] fue uno de los primeros sistemas y una contribución importante para los sistemas de localización en interiores. Fué desarrollado por el laboratorio de investigación de Olivetti (actualmente AT&T Cambridge). Necesita un punto de acceso de infrarrojos en cada habitación y el dispositivo móvil a localizar lleva un emisor de infrarrojos (Figura 2.6) que emite GIDs (Globally unique identifiers) cada 15 segundos. Teniendo en cuenta que las señales se reflejan en casi todas las superficies, las señales del GID se contienen dentro de cada habitación obteniéndose una localización con este nivel de granularidad. Al actualizarse la información cada 15 segundos, si el individuo se mueve a gran velocidad, pueden cometerse grandes errores. La información de localización se recopila en una base de datos centralizada en un servidor. Una estimación de localización de gran precisión resultaría complicada y requeriría múltiples antenas direccionales. Además, la escalabilidad en el sistema Active Badge es bastante pobre debido al rango limitado de los IR y no funciona correctamente en presencia de rayos de sol directos, siendo esto un problema en habitaciones con ventanas.



Figura 2.6: Active Badget Tags. Imagen extraída de [WHFG92]

2.2.1.2. Sistema basados en ultrasonidos (US)

Dependiendo de la densidad de la infraestructura y del grado de calibración, los sistemas basados en ultrasonidos pueden alcanzar precisiones entre unos pocos metros y centímetros, pero al requerir línea de visión entre los nodos necesitan un despliegue denso para dar grandes coberturas.

El sistema Cricket [PCB00] es un sistema de localización descentralizado, privado, con granularidad a nivel de habitación y de bajo coste (Figura 2.7). Cada componente del sistema, ya sea fijo o móvil, se configura independientemente, no se emplea ningún elemento central para registrar o sincronizar los diferentes elementos. La privacidad del usuario se mantiene permitiendo a los elementos móviles que estimen su propia localización de forma local sin comunicación con el exterior. Cricket permite saber sólo en qué habitación se encuentra pero lo consigue con una precisión del 95 %. Como cada baliza se configura de forma independiente del resto existe el peligro de que las balizas interfieran unas con otras. Para intentar evitarlo se introduce un tiempo aleatorio entre la emisión de las señales que van a emitir. Por otro lado, teniendo en cuenta que solo requiere un nodo estacionario de referencia por habitación el coste asociado a poder escalar la red es razonable.

Cricket estima las distancias con el TDOA de las señales RF y de ultrasonidos sincronizadas. Cada baliza emite un pulso RF que identifica inequívocamente el espacio en el que se encuentra. Los dispositivos móviles calculan la distancia recorrida por las señales captadas de las balizas.

Cricket Compass [PMBT01] es una extensión de Cricket que permite obtener tanto las coordenadas de localización como una estimación de la orientación. Continúa usando TDOA entre las señales sincronizadas de RF y ultrasonidos pero requiere que las balizas estén configuradas con coordenadas precisas en lugar de solamente identificadores de habitación como en el sistema Cricket. Se ha validado el modelo con experimentos obteniendo errores menores de 3° para ángulos de hasta 10°, y errores de hasta 15° cuando los ángulos son menores de 40°. Además el sistema demostró ser sensible a las reflexiones

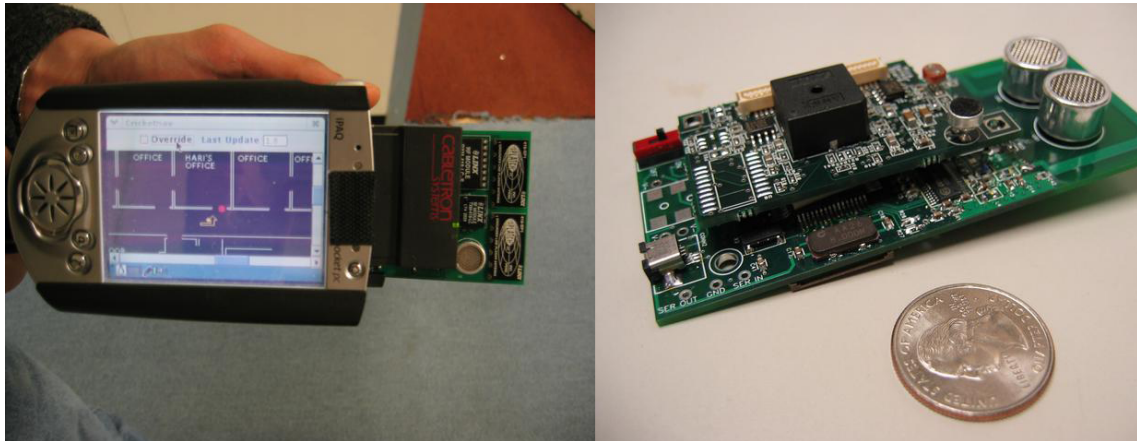


Figura 2.7: Imagen del receptor (izquierda) y de una baliza emisora (derecha) en el sistema Cricket. Imagen extraída de [PMBT01]

de ultrasonidos producidas en las paredes, ya que mientras que los errores de localización en el centro de una habitación son de 6 cm, si el móvil se acerca a la pared el error se incrementa hasta 25 cm.

El sistema Active Bat [WJH97] es una continuación del sistema Active Badge. Está construido sobre el mismo modelo de sistema centralizado de rastreo de localización que obtiene los datos de las balizas móviles. En Active Bat (Figura 2.8) los pequeños dispositivos activos, Bats, emiten periódicamente pulsos estrechos de ultrasonidos. Los ultrasonidos son detectados por los sensores que emplean la información de TOA para multilaterar empleando una minimización por mínimos cuadrados y obtener la localización del Bat. Este proceso tiene una precisión de 3 cm el 95 % del tiempo.

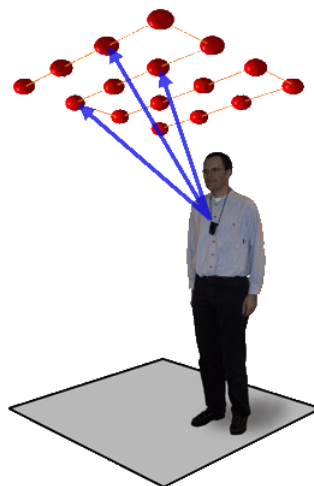


Figura 2.8: Sistema Active Bat. Imagen extraída de: <http://www.cl.cam.ac.uk>

Al igual que Cricket, Bat emplea el TDOA de las señales de ultrasonidos sincronizadas mediante un emisor RF y de la misma forma que hace Cricket Compass, usa multilateración para calcular las coordenadas de los elementos móviles en relación a los nodos de referencia. Sin embargo en este sistema se tomaron alternativas de diseño diferentes

a las tomadas en el sistema Cricket: las unidades situadas en las paredes son receptores de ultrasonidos para captar las señales de los Bats y están conectados a través de una red cableada y organizados jerárquicamente. Además, para disminuir la saturación del canal de ultrasonidos, cada Bat se registra en el sistema cada vez que entra en un área. Periódicamente la estación base envía los identificadores GID y simultáneamente envía pulsos de sincronización por la red cableada a todos los receptores. De este modo, cuando un Bat escucha el mensaje GID responde con una señal de ultrasonidos y el receptor es capaz de calcular el TDOA entre el pulso de sincronismo y la recepción de la señal de ultrasonidos. El sistema Active Bat está diseñado para colocar los tags dentro de una habitación y poder obtener un alto grado de precisión, pero es susceptible a los efectos de multicamino y reflexiones en el receptor.

2.2.1.3. Sistemas Ultrawideband (UWB)

Los emisores y receptores de UWB han sido empleados recientemente para estimar la localización en interiores, con precisiones por debajo de los 15 cm. En este tipo de sistemas pueden usarse tanto técnicas de tiempo de vuelo (TOF) como de medida de la amplitud de la señal (RSSI). Aunque el equipamiento para medir el RSSI está muy extendido y sale más económico, la precisión que se obtiene es menor que si se emplean sistemas basados en TOF, TOA o TDOA.

Ubisense [SG05] es un sistema de localización que está disponible comercialmente. El sistema de localización UWB de Ubisense consiste principalmente de tres componentes: tags alimentados con baterías que transmiten pulsos UWB que sirven para determinar la localización (Ubitags), sensores alimentados a través de la red eléctrica (Ubisensors) fijados sobre la infraestructura y encargados de recibir y evaluar las señales de los Ubitags, y la plataforma software que agrega los datos de localización, los analiza, los presenta y comunica la información a los usuarios y sistemas de información relevantes. Los Ubisensors, son puntos de referencia fijos que generalmente se colocan en lugares con línea de visión como por ejemplo en los techos, y se conectan con el maestro por Ethernet para configurar las celdas de Ubisense llamadas Ubicells.

La localización se basa en las medidas de TOA y TDOA de los pulsos UWB de los Ubitags a los Ubisensores. Los Ubisensores se agrupan por celdas, que por lo general suelen ser de forma rectangular, y en cada una de ellas existe un sensor maestro que se encarga de coordinar y comunicarse con todos los tags y resto de sensores. La información de localización se puede enviar tanto por cable Ethernet como por wireless LAN al software del motor de localización que se encarga de agregar los datos y enviarlos a través de un API a un programa externo o a la plataforma de localización de Ubisense para su visualización y procesamiento espacial (Figura 2.9).

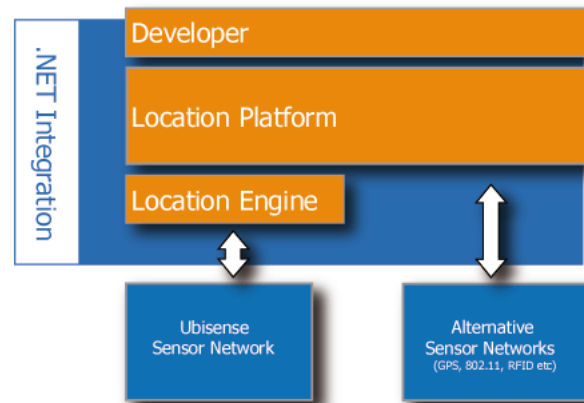


Figura 2.9: Sistema Ubisense

Existen varias opciones para diseñar sistemas que se adecuen a las necesidades de cada aplicación y al entorno físico en el que se requiera la implantación de la plataforma. Los parámetros típicos son la geometría del área y los materiales que hay presentes, la precisión requerida, qué objetos tendrán tags incorporados y la velocidad con la que se moverán, duración de la batería deseada, presencia de otros sistemas RF... Los tags son parte de una red de sensores que se despliega a lo largo del área que se quiere monitorizar y se utiliza la infraestructura de red y protocolos estándar para comunicar los dispositivos.

2.2.1.4. Sistemas basados en marcadores

Una forma de estimar la posición del usuario y su orientación es el empleo de indicadores visuales pegados en los techos o en las paredes (marcadores). El despliegue de la infraestructura de localización resulta relativamente barato y es posible conocer la posición y orientación del usuario de forma robusta ya que el reconocimiento de indicadores en imágenes capturadas con cámaras no es difícil. La desventaja que presenta este método es que, a veces, los indicadores pueden llegar a ser demasiado vistosos, exagerados, y pueden llamar la atención más de la cuenta en un determinado entorno (Figura 2.10), causando un efecto visual indeseado [NF02].

Por otro lado está la posibilidad de emplear indicadores invisibles o diseñados para mezclarse con el entorno de modo que se obtenga la localización sin perturbar el mismo como sucedía en el caso anterior (Figura 2.11).

En [NKY08] se propone un sistema de localización empleando indicadores invisibles. Se trata de un método que tiene dos fases, la primera es para la introducción en el sistema de los parámetros necesarios para identificar y posicionar los indicadores invisibles, la segunda se centraría en la instalación del sistema en el entorno de localización (Figura 2.12).



Figura 2.10: Persona que viste un VIS-Tracker (vision-inertial self-tracker) en un entorno plagado de indicadores. Imagen extraída de [NF02]



Figura 2.11: Vista de la escena (izquierda) y vista de IR (derecha). Imagen extraída de [NKY08].

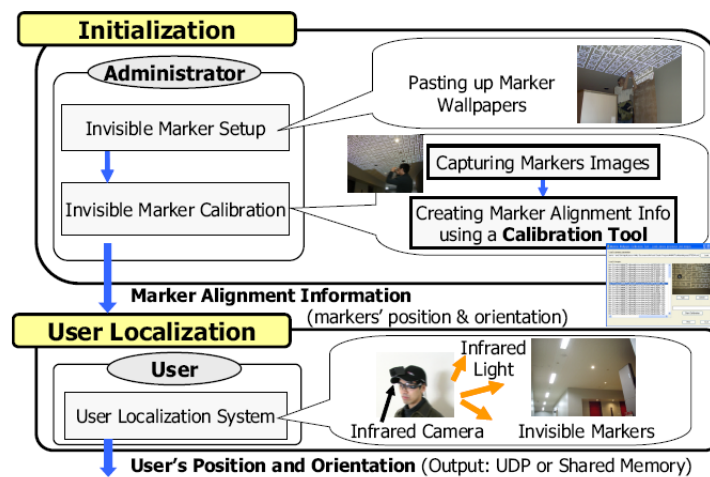


Figura 2.12: Resumen del sistema de localización de usuarios empleando indicadores invisibles. Imagen extraída de [NKY08].

El usuario va equipado de un ordenador portable y una cámara de IR que captura indicadores invisibles por medio del reflejo de los infrarrojos. La precisión de la posición y orientación estimadas empleando los indicadores invisibles llega a ser bastante elevada. En el experimento que se detalla, donde los indicadores tienen un área de 16 cm^2 y hay

una densidad de indicadores de 24 indicadores/ m^2 , el error cometido es de 13 mm con una desviación estándar de 12 mm.

2.2.1.5. Localizacion RFID

RFID (Radio Frequency Identification) es una tecnología inalámbrica capaz de identificar de forma única los objetos y transmitir esta información al sistema a través de señales radio. Para dar servicios de localización de buena precisión con esta tecnología se requiere un alto número de sensores. Los parámetros que hay que establecer son: la densidad de dispositivos RFID, el área de cobertura, y el diagrama de radiación de las balizas (Figura 2.13).

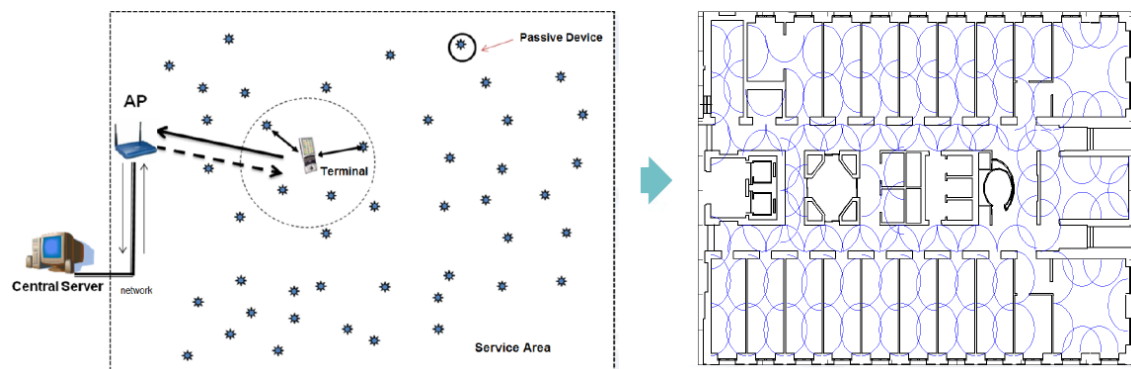


Figura 2.13: Arquitectura de un posible sistema RFID. Imagen extraída de [HVBW01]

Entre los diferentes algoritmos de posicionamiento mediante RFID, se puede realizar la siguiente clasificación: los de proximidad, que se basan únicamente en el identificador del tag más cercano; los que además de lo anterior pueden obtener medida de potencia de la señal RFID recibida y los que calculan el tiempo de vuelo de la señal.

Esta tecnología ha sido considerada como una solución de bajo coste para el seguimiento o rastreo de objetos en interiores. Algunas de las implementaciones utilizan tags activos (alimentados con baterías). Existen muchas otras implementaciones basadas en el uso de señales RFID, medidas de amplitud de señal o medidas de tiempo de vuelo [HVBW01, HBS08].

2.2.1.6. Localización con redes ZigBee

ZigBee es un estándar que define un conjunto de protocolos de comunicaciones para redes inalámbricas de RF, con baja tasa de transferencia de datos (250kbps como máximo); son muy apropiadas para construir Wireless Sensor Networks (WSN). Los algoritmos de localización basados en WSN estiman la localización a partir de la señal procedente de nodos cuya posición es conocida, realizando medidas de distancia o ángulo de incidencia. La mayoría de los algoritmos de localización basados en ZigBee usan sólo los niveles RSSI para estimar la localización debido a un requerimiento mínimo o nulo de

hardware adicional. El estándar ZigBee fue desarrollado por la ZigBee Alliance, formada en el año 2002. Se adoptó IEEE 802.15.4 como sus protocolos en sus capas física y MAC (Media Access Control), de modo que un dispositivo compatible con ZigBee también lo es con IEEE 802.15.4 a estos niveles. En general, la precisión que se puede obtener con ZigBee depende de la densidad de balizas incorporadas en el sistema, luego es escalable; en cuanto a la computación de posición, ésta se podrá realizar tanto de forma distribuida como centralizada.

Una de las ventajas que presenta ZigBee para desarrollar sistemas de localización, es la facilidad de obtener los niveles RSSI ya que viene incorporada con cada uno de los paquetes enviados y no hace falta hardware adicional para ello. Si en lugar de trabajar con niveles de señal se pretende determinar el TOA se requiere un reloj de alta precisión y para calcular el AOA (Angle of Arrival) hacen falta modificaciones hardware. Ambas soluciones aumentarían el coste del equipo.

En [PB07] se realiza un análisis de cada una de las capas que componen el protocolo ZigBee y los diferentes métodos de localización.

2.2.2. Localización independiente de la estructura

Para poder evitar los costes asociados al despliegue de una infraestructura específica, este tipo de sistemas reutilizan alguna infraestructura apropiada ya existente. Estas técnicas de localización tienen la ventaja de ser soluciones eficientes en costes donde se pueden usar dispositivos abiertos, no propietarios, que cumplan el estándar seleccionado para dar servicios de localización apropiado. Las primeras soluciones fueron las basadas en puntos de acceso Wi-Fi, [BP00], y poco después se comenzó a explorar la localización con Bluetooth [FMP05] y los basados en sistemas de telefonía como GSM [OVLdL05, VDH⁺07] o torres de transmisión de emisoras FM.

Existen sistemas que combinan diferentes tecnologías como por ejemplo Redpin [Bol08], que analiza el nivel de señal de GSM, Bluetooth y Wi-Fi en un dispositivo móvil (Figura 2.14)

2.2.2.1. Localización WiFi

La mayoría de las soluciones de localización con Wi-Fi están basadas en técnicas de fingerprinting. Estas técnicas tienen las ventajas de su sencillez y rapidez de despliegue, la ausencia de equipamiento adicional a la de una red LAN y su coste asociado.

Los errores de posicionamiento debidos a estos sistemas son consecuencia de: el limitado número de puntos de acceso y su emplazamiento; la granularidad del mapa radio (distancia de muestreo); errores en las medidas, esto es, diferentes interpretaciones con diferentes dispositivos; y los cambios impredecibles de las características de propagación

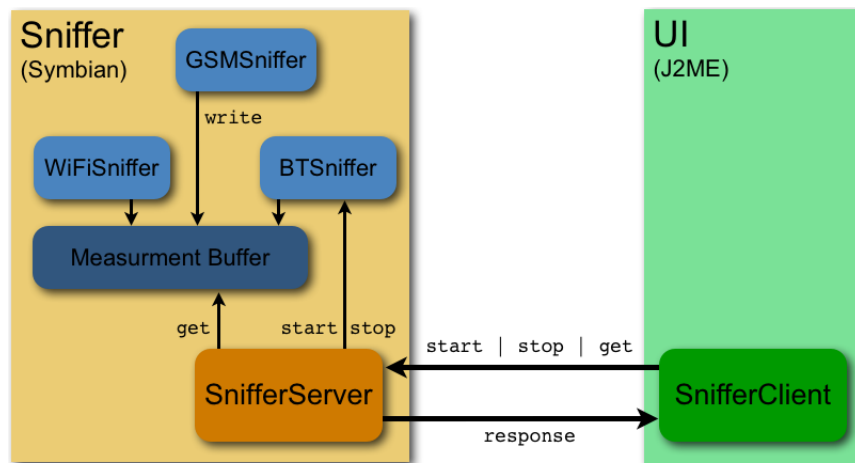


Figura 2.14: Redpin: arquitectura del “sniffer” (lector de paquetes) del teléfono móvil. Imagen extraída de [Bol08]

del entorno. Uno de los primeros sistemas de este tipo desarrollados es el sistema RADAR [BP00].

La triangulación Wi-Fi evita los requerimientos de línea de visión pero sufre los efectos del multicamino debido a las reflexiones en el entorno. Usando fingerprinting se mejoran las estimaciones teniendo en cuenta los efectos que causan los edificios, los objetos y la gente, así como las reflexiones y atenuación. Aun así no hay que olvidar que los cambios en el medio causan importantes fluctuaciones en las señales Wi-Fi en un mismo punto a medida que pasa el tiempo, de modo que las medidas tomadas en la fase de entrenamiento se quedan obsoletas con rapidez.

Empleando cuatro puntos de acceso Wi-Fi, RADAR [BP00] localiza un portátil con una resolución media de 2-3 metros. Es un sistema que localiza y es capaz de seguir a los individuos dentro de un edificio. Funciona recogiendo y procesando niveles de señal de diferentes estaciones base colocadas de forma que se superpongan las áreas de cobertura dentro de la zona de interés. Combina las medidas de una fase de inicialización con un modelo simple de propagación de señales para determinar la localización. Desde la primera versión ha habido muchas mejoras del algoritmo de fingerprinting empleado para mejorar su precisión.

Una de las soluciones comerciales de localización más famosas que emplea Wi-Fi es de la empresa Ekahau [eka11]. La aplicación EPE (Ekahau Positioning Engine) se encarga de estimar la localización. Incluye diversos algoritmos y métodos patentados para calcular una localización precisa en tiempo real. Para poder funcionar correctamente EPE necesita conocer el área de cobertura de cada punto de acceso Wi-Fi y las características y patrones del entorno radioeléctrico en el que se va llevar a cabo la localización. Esta información se obtiene mediante la Ekahau Site Survey Tool.

Por otro lado los métodos basados en TOA son una alternativa para hacer frente a las restricciones que presentan las soluciones de fingerprinting en Wi-Fi, sobre todo en as-

pectos de precisión, estabilidad, y rapidez de despliegue. La escalabilidad de los métodos basados en TOA a veces se ve afectada al aumentar el número de usuarios por el tráfico generado en el proceso de posicionamiento. No se ha propuesto formalmente casi ningún sistema basado en TOA, ni en la literatura científica, ni como productos privados, ya que dicha técnica presenta ciertos retos que hay que solventar. No hay que olvidar que uno de los puntos clave para poder obtener una gran precisión con TOA es que los relojes no deben tener imprecisiones, ya que un error de un microsegundo podría introducir un error del rango de trescientos metros, además se necesita un ancho de banda elevado. En la actualidad existen algunas aproximaciones que tratan de implementar esta tecnología [GB07, CBAC09].

2.2.2.2. Localización GSM

La mayor parte de los trabajos sobre localización en telefonía móvil se ha concentrado en los sistemas GSM (Global System for Mobile Communication) utilizando técnicas fingerprinting basadas en RSSI. Se trata de una tecnología para dispositivos móviles ampliamente extendida. En [OVLdL05] se empleó GSM para la localización en interiores. Además de usar las seis celdas con mayor señal para el fingerprinting, también incluyeron aquellas celdas con suficiente nivel como para ser detectadas. Consiguieron una precisión comparable a los sistemas basados en Wi-Fi a partir de una granularidad de 1.5 metros. La continuación del trabajo anterior se encuentra en [VDH⁺07] donde presentan las experiencias y resultados obtenidos a partir de dos casos: en el primero analizan cómo afecta el ancho de banda y la selección del canal a la hora de estimar la localización dentro de una planta, con su sistema obtuvieron precisiones en el rango de 1.94 a 4.07 metros; en el segundo estudio analizaron la capacidad de distinguir con la tecnología GSM en qué planta se encuentra un individuo dentro de un edificio grande con varias plantas, el sistema identificó la planta correctamente en el 60 % de los casos y se quedó entre dos plantas en el 98 %.

De forma similar, SkyLoc [VLHdL07] emplea GSM para poder detectar dónde se encuentra un móvil dentro de un edificio grande de varias plantas. El sistema está implementado en C# y fue testado en un móvil AudioVox SMT 5600 con Windows Mobile 2003. SkyLoc está compuesto por dos componentes: una aplicación para recoger datos, PlaceLogger, y una aplicación para realizar y visualizar el resultado de la localización, SkyLoc. Este sistema consigue identificar la planta correcta del usuario en el 73 % de los casos y se queda entre dos plantas en el 97 % de los casos

2.2.2.3. Localización con Bluetooth

Existen soluciones de localización que realizan las estimaciones de posición analizando las señales RF de dispositivos móviles equipados con Bluetooth (teléfonos móviles,

PDA's, ordenadores, etc.). La mayoría de las soluciones de localización de este tipo estiman la localización a partir de características de la red como el RSSI. En Bluetooth la señal recibida puede caracterizarse mediante los parámetros de RSSI y LQ (Link Quality), sin embargo en Bluetooth el RSSI no es una medida tan fiable como en el caso de ZigBee debido a las heterogeneidades existentes en el hardware de los dispositivos Bluetooth. Esto se debe en parte a la imprecisión en la definición del parámetro RSS en el estándar Bluetooth. Un sistema de localización basado en RSS Bluetooth se encuentra en [FMP05].

2.3. Sistemas de Autocalibración para LPS interiores

Como se ha mencionado anteriormente normalmente los sistemas de localización basados en la medida de distancias a balizas o puntos de referencia, necesitan conocer la posición exacta de las balizas para después poder emplear algún algoritmo de trilateración. Por eso es necesario que previamente se haya medido o calculado la distancia de cada baliza al origen de referencia y se hayan introducido éstas al sistema. Este hecho hace que este tipo de sistemas sean poco portables y adaptables ya que el proceso de calibración suele requerir un cierto tiempo y varias personas tomando medidas.

En este apartado se analizan las técnicas que actualmente se emplean para poder obtener de forma más automática la posición de las balizas. En una primera sección se describirán los diferentes mecanismos que existen para recopilar la información dentro del sistema. En la segunda sección se describirán alguno de los algoritmos desarrollados para obtener la posición de las balizas.

2.3.1. Métodos de obtención de la información

Para poder obtener la posición de las balizas es necesario obtener información (medidas) en varios puntos del entorno. Existen varios métodos para realizar este procedimiento:

- Basados en personas: una persona equipada con un dispositivo de localización va caminando por la habitación y el sistema va recopilando la información.
- Basados en robots móviles: el sistema de recopilación de información está en un dispositivo móvil que se desplaza, normalmente por el suelo.
- Basados en patrones: el dispositivo de calibración se coloca en ciertos puntos fijos conocidos de un patrón para realizar la calibración.

2.3.2. Autocalibración empleando posicionamiento inverso

Uno de los métodos más usados que existen para obtener la posición de las balizas en un sistema LPS, consiste en emplear el posicionamiento inverso, esto es, tomar varias medidas de distancia en puntos conocidos y luego obtener la posición de las balizas empleando el algoritmo de localización. En [MF99] se implementa un sistema que emplea este tipo de autocalibración. Para obtener la posición de las balizas se mueve el transmisor al menos a cinco puntos conocidos y en cada uno de ellos se obtiene el tiempo de vuelo a las balizas.

Algunos de los inconvenientes de estos sistemas son que, o bien se necesita un dispositivo de calibración muy grande que sea capaz de detectar todas las balizas del sistema, o bien, si el dispositivo de calibración es pequeño se debe situar muy lejos de las balizas para poder detectarlas todas (Figura 2.15). Esto en muchos casos no es viable ya que puede no haber espacio suficiente en habitaciones pequeñas y también se producen fenómenos de doble rebote. Además si se necesita conocer la posición de muchos puntos de medida, puede ser en ocasiones más sencillo medir la posición de las balizas respecto al sistema de referencia de forma manual.

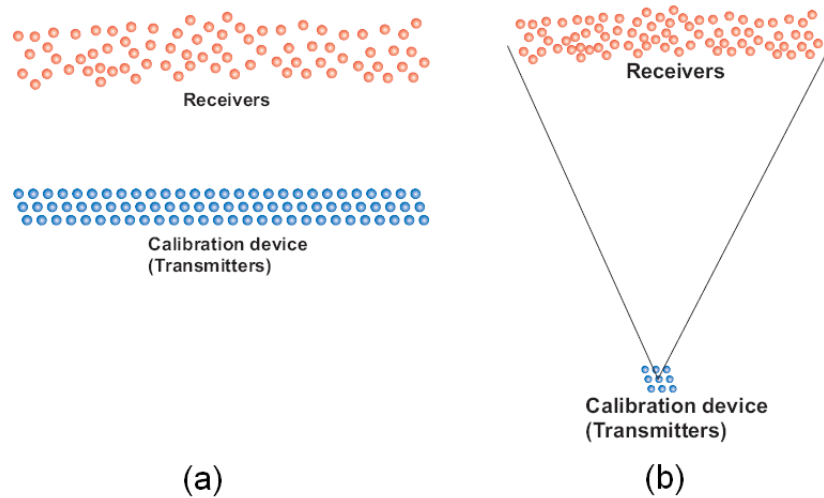


Figura 2.15: Problema de posicionamiento inverso. (a) Dispositivo de calibración muy grande. (b) Dispositivo de calibración muy alejado. Imagen extraída de [NNHM04]

En [NNHM04] se desarrolla otro método basado en posicionamiento inverso que trata de solucionar algunos de los problemas del sistema anterior. Este sistema emplea un dispositivo de calibración con tres o más transmisores en posiciones fijas y realiza el siguiente proceso (Figura 2.16):

1. Se mueve el dispositivo de calibración a múltiples posiciones de la habitación (A, B y C en la figura 2.16).
2. Se calcula la posición de las balizas detectadas en cada posición seleccionada, suponiendo que el origen está en el dispositivo de calibración.

3. Se seleccionan las balizas que son detectadas desde dos o más posiciones diferentes.
4. De todos los puntos elegidos se selecciona uno como sistema de coordenadas global. A partir de las balizas seleccionadas en el paso 3 se calculan las matrices de transformación entre los diferentes sistemas de referencia locales.
5. Se calculan las posiciones de las balizas respecto al sistema de referencia global.

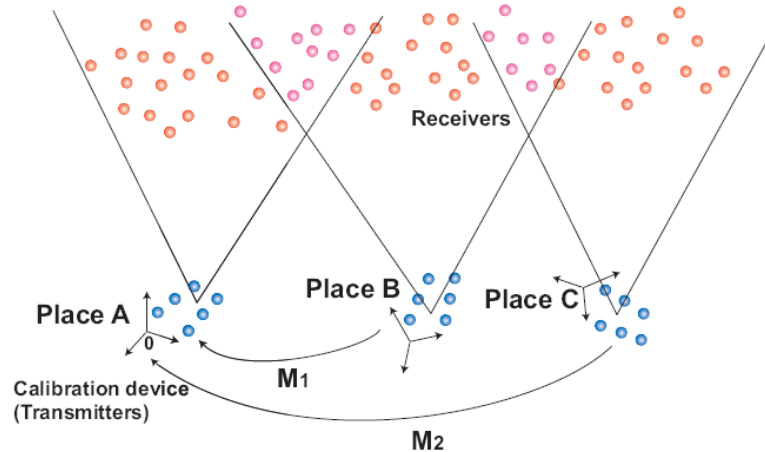


Figura 2.16: Método de calibración. Imagen extraída de [NNHM04]

2.3.3. Sistemas de Autocalibración basados en métodos iterativos

El principal problema de los métodos de autocalibración usando posicionamiento inverso es, como se ha mencionado anteriormente, que normalmente se necesita conocer de manera precisa varios puntos de medida, por lo que muchas veces resulta más útil medir directamente la posición de las balizas. Los métodos iterativos tratan de solucionar este problema.

En [DM03] se presenta un método basado en trilateración esférica que obtiene la posición de las balizas tomando medidas desde puntos cuya posición es conocida y otros en los que se desconoce su posición. El algoritmo supone una posición inicial tanto de las balizas como de los puntos de medida cuya posición es desconocida y aplica un algoritmo Levenberg-Marquardt para minimizar el error entre la distancia obtenida con las balizas estimadas y las distancias reales. Como el espacio de búsqueda puede ser muy grande, el algoritmo se repite 100 veces con valores aleatorios y utiliza funciones de ajuste para refinar los resultados. Los pasos seguidos por el algoritmo son:

1. Obtener las distancias en diferentes puntos.
2. Elegir subgrupos con diferentes valores iniciales de las incógnitas dentro del espacio de búsqueda.

3. Para cada subgrupo:

- Encontrar 100 soluciones de baja precisión.
- Seleccionar las seis mejores soluciones con las funciones de comparación.
- Refinar para obtener tres soluciones con moderada precisión.
- Refinar para obtener la mejor solución (alta precisión).

4. Obtener la mejor solución de todos los subgrupos.

Posteriormente en [DMC⁺05] se adapta el método anterior para aplicarlo con trilateración hiperbólica.

En [Kur04] se presenta un método de autocalibración para un sistema LPS que combina la información obtenida por la odometría del robot con la información de distancia proporcionada por un LPS mediante un filtro de Kalman extendido para tratar de averiguar la posición de las balizas que componen el LPS. Este método consigue precisiones en torno a centímetros

2.4. Objetivos planteados

Como se ha mostrado a lo largo de este capítulo, la localización en interiores es un problema todavía no resuelto al que se aplican multitud de tecnologías, esquemas y métodos de cómputo. Una gran parte de los sistemas existentes instalan balizas en el entorno donde se quiere localizar al nodo móvil y, a través de las posiciones de las mismas y de las distancias medidas al móvil, obtienen la posición de dicho nodo. La mayor parte de los algoritmos de localización propuestos suponen que la posición de las balizas respecto al origen de referencias es conocida por el sistema, pero en la práctica obtener dicha posición con exactitud es bastante tedioso y complicado y suele requerir bastante tiempo, haciendo que estos sistemas sean poco portables y adaptables.

Además como queda demostrado en [TR05] los errores cometidos en la localización de las balizas afectan a la localización de igual forma que un error cometido en la medida de distancia haciendo muy importante la correcta localización de las mismas. Por ejemplo, se podría tener un sistema en el que se miden tiempos de vuelo con mucha precisión (del orden de magnitud de picosegundos) pero, si en la localización de las balizas se tienen errores del orden de decímetros, en la localización del nodo móvil se tendrán errores superiores a decímetros.

Teniendo en cuenta estas consideraciones, el objetivo global de la tesis es obtener un método de calibración que permita obtener de manera semi-automática las posiciones de las balizas que componen un sistema de posicionamiento local. Este objetivo global conlleva una serie de objetivos parciales que se detallan a continuación:

- *Obtención de un algoritmo de posicionamiento hiperbólico más eficiente que el método de resolución directa por medio de Gauss-Newton .*

Actualmente el método de resolución directa de las ecuaciones de Gauss-Newton es uno de los métodos de posicionamiento más extendido debido a la precisión que ofrece [YTBJ08]. Uno de sus inconvenientes es que, al ser un método iterativo, su tiempo de cómputo puede ser elevado si se buscan altas precisiones en la localización del nodo móvil. También los tiempos de cómputo de este algoritmo se disparan en el caso de emplear trilateración hiperbólica. Por eso en la primera fase de esta tesis se busca un nuevo algoritmo de localización que sea más eficiente a la hora de realizar la localización, esto es que tenga similares propiedades que el método de resolución directa por Gauss-Newton pero con tiempos de cómputo muy inferiores. Para ello se emplearán las propiedades proporcionadas por los bideterminantes de Cayley-Menger [TORC03]

- *Propuesta de un sistema de calibración para un sistema LPS individual*

Como se ha mostrado en la sección 2.3.2 de este capítulo, una de las formas más comunes de obtener las posiciones de las balizas de un sistema, consiste en aplicar posicionamiento inverso a partir de medidas a las balizas desde diversos puntos cuya ubicación sea conocida. En esta tesis se buscarán nuevos algoritmos que permitan realizar la calibración del LPS, sin tener que conocer la localización de todos los puntos de medida; de esta forma se puede tener un dispositivo fijo y de pequeño tamaño, desde donde se tomarían las medidas de los puntos conocidos y después se realizarían medidas por diferentes zonas del área de cobertura, en ubicaciones desconocidas para el sistema. Además se analizará si es posible calcular, conociendo la estructura del sistema (no su ubicación), las mejores zonas donde situar los puntos de test desconocidos para que el proceso de autocalibración cometa el menor error posible.

- *Método de calibración de un sistema LPS global compuesto por varios LPS independientes*

Con el sistema de calibración anterior se pueden calibrar varios LPS de forma local, esto es, cada uno respecto a su propio sistema de referencia. Un objetivo fundamental de la tesis consiste en obtener un algoritmo que sea capaz de calibrar todos los LPS independientes respecto a un único sistema de referencia global. Para ello se propondrá un método que empleando un robot móvil sea capaz de fusionar la información de odometría proporcionado por el mismo y la información de distancia proporcionada por los LPS.

- *Validación de los métodos propuestos mediante la realización de pruebas bajo simulación y experimentales en un LPS ultrasónico*

Finalmente, se llevará a cabo un conjunto de simulaciones y pruebas experimentales en un LPS real con objeto de validar el comportamiento de los diferentes métodos propuestos

en la tesis. Se evaluarán diferentes aspectos tales como la precisión en las medidas, la estimación de la posición de las balizas en las pruebas de autocalibración, y el efecto en la calibración y en el posicionamiento según el tipo de trilateración: esférica o hiperbólica.

Algoritmo de posicionamiento basado en los determinantes de Cayley-Menger

En este capítulo se presenta un nuevo algoritmo de posicionamiento basado en el operador matemático de Cayley y Menger descrito en [Som58]. Este operador presenta unas propiedades geométricas que permiten encontrar la ubicación de un nodo móvil de una forma más rápida e igual de precisa que los métodos iterativos basados en la resolución directa de las ecuaciones de distancia empleando el algoritmo de Gauss-Newton [YTB08] o similares.

El capítulo se ha estructurado de la siguiente manera: en la primera sección se define qué es un bideterminante de Cayley-Menger y cuales son sus propiedades geométricas; a continuación se explica como emplear este operador para poder realizar un posicionamiento esférico; en la sección siguiente se proponen diversos algoritmos en el caso de que se utilice un posicionamiento hiperbólico; posteriormente se realizan varias pruebas bajo simulación para comprobar las propiedades de los algoritmos propuestos; por último, se presentan las conclusiones extraídas de este capítulo.

3.1. Bideterminantes de Cayley-Menger

En esta sección se describe el operador matemático bideterminante de Cayley-Menger y se analizan algunas de sus propiedades más importantes, las cuales son utilizadas en secciones posteriores para la propuesta de nuevos algoritmos de trilateración hiperbólica.

Según [Som58] el bideterminante de Cayley-Menger de dos secuencias de n puntos $[\mathbf{p}_1, \dots, \mathbf{p}_n]$ y $[\mathbf{q}_1, \dots, \mathbf{q}_n]$ se define como :

$$D(\mathbf{p}_1, \dots, \mathbf{p}_n; \mathbf{q}_1, \dots, \mathbf{q}_n) = 2 \left(\frac{-1}{2} \right)^n \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & D(\mathbf{p}_1, \mathbf{q}_1) & D(\mathbf{p}_1, \mathbf{q}_2) & \cdots & D(\mathbf{p}_1, \mathbf{q}_n) \\ 1 & D(\mathbf{p}_2, \mathbf{q}_1) & D(\mathbf{p}_2, \mathbf{q}_2) & \cdots & D(\mathbf{p}_2, \mathbf{q}_n) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & D(\mathbf{p}_n, \mathbf{q}_1) & D(\mathbf{p}_n, \mathbf{q}_2) & \cdots & D(\mathbf{p}_n, \mathbf{q}_n) \end{vmatrix} \quad (3.1)$$

Donde $D(\mathbf{p}_i, \mathbf{q}_j)$ es el cuadrado de la distancia Euclídea entre el punto \mathbf{p}_i y el punto \mathbf{q}_j . Si las dos secuencias de puntos son iguales se puede abreviar el término $D(\mathbf{p}_1, \dots, \mathbf{p}_n; \mathbf{p}_1, \dots, \mathbf{p}_n)$ por $D(\mathbf{p}_1, \dots, \mathbf{p}_n)$. En este caso, a este término se le conoce como determinante de Cayley-Menger.

3.1.1. Propiedades geométricas del operador de Cayley-Menger

La principal propiedad de los determinantes de Cayley-Menger es que su valor es igual a $((n-1)!)^2$ veces el cuadrado del hiper-volumen formado por la secuencia de puntos $\mathbf{p}_1, \dots, \mathbf{p}_n$ en \mathbf{R}^{n-1} [Usp48]. En la figura 3.1 se pueden observar varios ejemplos de esta propiedad. Para $n=2$ el determinante equivale al cuadrado de la distancia euclídea entre los puntos. Para $n=3$ equivale a cuatro veces el área al cuadrado del triángulo formado por los puntos. Por último para $n=4$ el determinante de Cayley-Menger equivaldría a 36 veces el volumen al cuadrado del tetraedro formado por los puntos y así sucesivamente.

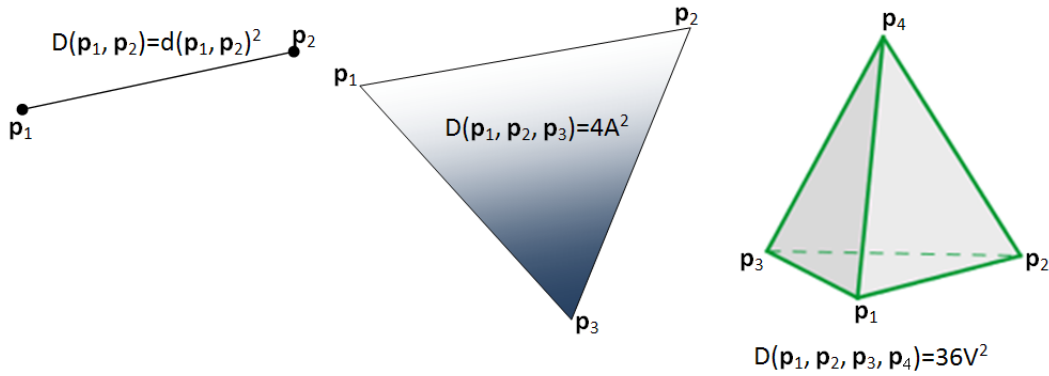


Figura 3.1: Propiedad geométrica de los determinantes de Cayley-Menger.

En el caso de dos secuencias de dos puntos el bideterminante se puede expresar como:

$$\begin{aligned}
 D(\mathbf{p}_1, \mathbf{p}_2; \mathbf{q}_1, \mathbf{q}_2) &= \frac{1}{2} \begin{vmatrix} 0 & 1 & 1 \\ 1 & D(\mathbf{p}_1, \mathbf{q}_1) & D(\mathbf{p}_1, \mathbf{q}_2) \\ 1 & D(\mathbf{p}_2, \mathbf{q}_1) & D(\mathbf{p}_2, \mathbf{q}_2) \end{vmatrix} \\
 &= \frac{1}{2} (D(\mathbf{p}_2, \mathbf{q}_1) + D(\mathbf{p}_1, \mathbf{q}_2) - D(\mathbf{p}_2, \mathbf{q}_2)) \\
 &= (\mathbf{p}_1 - \mathbf{p}_2) \cdot (\mathbf{q}_1 - \mathbf{q}_2) = \left(\sqrt{D(\mathbf{p}_1, \mathbf{p}_2) D(\mathbf{q}_1, \mathbf{q}_2)} \right) \cdot \cos(\theta)
 \end{aligned} \tag{3.2}$$

Por lo que el bideterminante es igual al producto de la distancia entre los puntos de cada secuencia por el coseno del ángulo que existe entre las líneas definidas por las dos secuencias.

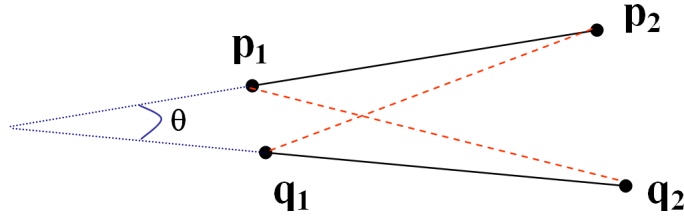


Figura 3.2: Propiedad geométrica de los bideterminantes de Cayley-Menger para secuencias de dos puntos.

En el caso de que el primer punto sea el mismo en las dos secuencias, (véase figura 3.3), se puede obtener el ángulo θ entre las rectas formadas por las dos secuencias como:

$$\cos(\theta) = \frac{D(\mathbf{p}_1, \mathbf{p}_2; \mathbf{q}_1, \mathbf{q}_2)}{\sqrt{D(\mathbf{p}_1, \mathbf{p}_2) D(\mathbf{q}_1, \mathbf{q}_2)}} \quad (3.3)$$

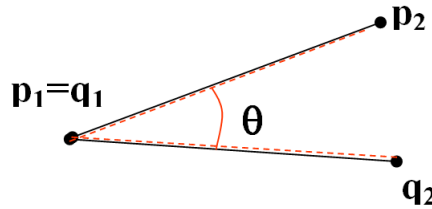


Figura 3.3: Ángulo entre dos rectas

En el caso de secuencias formadas por tres puntos, el bideterminante se puede expresar en la forma:

$$\begin{aligned} D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) &= -\frac{1}{4} \begin{vmatrix} 0 & 1 & 1 & 1 \\ 1 & D(\mathbf{p}_1, \mathbf{q}_1) & D(\mathbf{p}_1, \mathbf{q}_2) & D(\mathbf{p}_1, \mathbf{q}_3) \\ 1 & D(\mathbf{p}_2, \mathbf{q}_1) & D(\mathbf{p}_2, \mathbf{q}_2) & D(\mathbf{p}_2, \mathbf{q}_3) \\ 1 & D(\mathbf{p}_3, \mathbf{q}_1) & D(\mathbf{p}_3, \mathbf{q}_2) & D(\mathbf{p}_3, \mathbf{q}_3) \end{vmatrix} \\ &= [(\mathbf{p}_1 - \mathbf{p}_3) \times (\mathbf{p}_2 - \mathbf{p}_3)] \cdot [(\mathbf{q}_1 - \mathbf{q}_3) \times (\mathbf{q}_2 - \mathbf{q}_3)] \\ &= 4A_1A_2\cos(\phi) \end{aligned} \quad (3.4)$$

Donde A_1 y A_2 son las áreas de los triángulos formados por los puntos $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ y $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ respectivamente, y ϕ es el ángulo existente entre los dos triángulos, formados por cada tripleta de puntos.

En el caso particular en que $\mathbf{p}_1 = \mathbf{q}_1$ y $\mathbf{p}_2 = \mathbf{q}_2$, la estructura resultante se presenta en la figura 3.4, formada por dos triángulos con lado común en la unión de los puntos comunes. En esta estructura el ángulo entre los dos triángulos puede expresarse como (figura 3.4) :

$$\cos(\phi) = \frac{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}{\sqrt{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3) D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}} \quad (3.5)$$

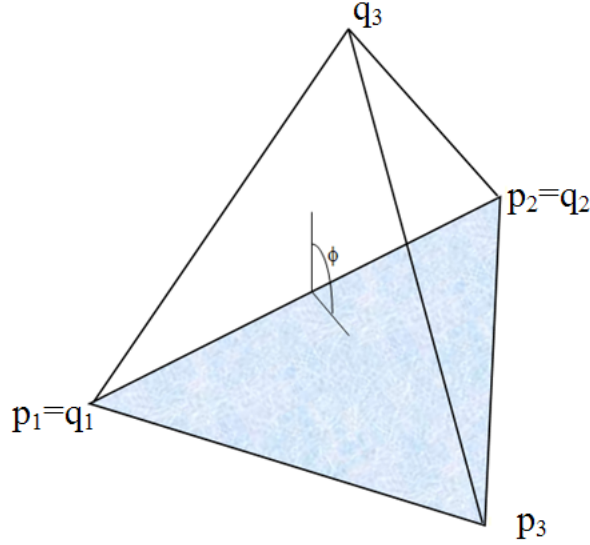


Figura 3.4: Ángulo definido por dos triángulos

En [TORC03] se pueden encontrar las demostraciones de estas propiedades y otras propiedades geométricas adicionales. En esta sección solo se han descrito aquellas que son útiles para los algoritmos de posicionamiento realizados en esta tesis.

3.2. Posicionamiento esférico

En esta sección se expone en primer lugar el algoritmo de trilateración esférica descrito en [TR05], para a partir de él poder obtener nuevos algoritmos pero esta vez adaptados al posicionamiento hiperbólico.

Dados tres puntos fijos conocidos en el espacio que actúan como balizas (\mathbf{q}_1 , \mathbf{q}_2 , \mathbf{q}_3) el problema de localización consiste en encontrar las coordenadas de otro punto, \mathbf{m} , conociendo la distancia desde los puntos conocidos y dicho punto. (l_1, l_2, l_3) . En la figura 3.5 se muestra un esquema de este problema.

El primer paso para obtener la posición de \mathbf{m} consiste en calcular las coordenadas de su proyección ortogonal, \mathbf{p} , a la base del triángulo formado por los tres puntos conocidos (\mathbf{q}_1 , \mathbf{q}_2 , \mathbf{q}_3).

Para poder obtener las coordenadas de este punto \mathbf{p} se van a utilizar las coordenadas baricéntricas descritas en [MLBD02] en donde se demuestra que la posición de un punto \mathbf{p} con respecto a los vértices de un polígono de N lados ($\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_N$) se puede expresar como:

$$\mathbf{p} \cdot \sum_{j=1}^N w_j = \sum_{j=1}^N (w_j \cdot \mathbf{q}_j) \quad (3.6)$$

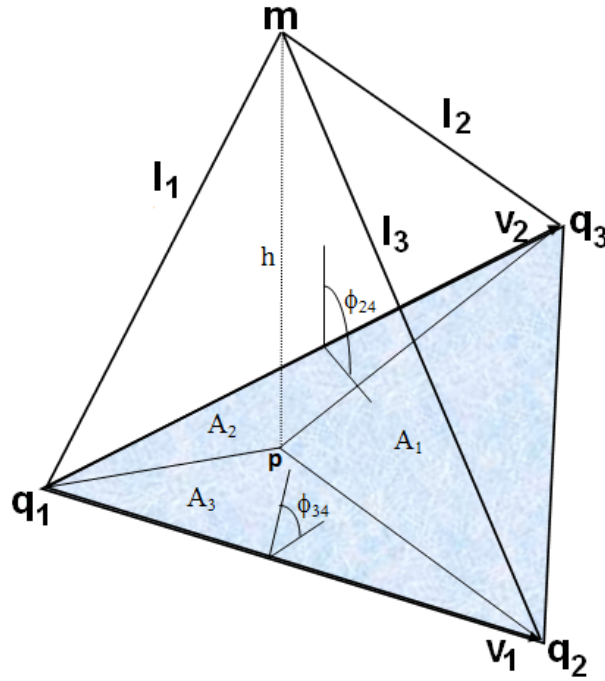


Figura 3.5: Problema de localización. Obtención de las coordenadas del punto \mathbf{m} a partir de las distancias l_1, l_2, l_3 a sendos puntos $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ de coordenadas conocidas.

Siendo w_j , unas constantes de ponderación que se obtienen a partir de la expresión:

$$w_j = A(\mathbf{q}_{j-1}, \mathbf{q}_j, \mathbf{q}_{j+1}) \cdot \prod_{k \notin \{j, j+1\}} A(\mathbf{q}_{k-1}, \mathbf{q}_k, \mathbf{p}) \quad (3.7)$$

Siendo $A(\mathbf{q}_{j-1}, \mathbf{q}_j, \mathbf{q}_{j+1})$ y $A(\mathbf{q}_{k-1}, \mathbf{q}_k, \mathbf{p})$ la áreas con signo del triángulo formado por los vértices $(\mathbf{q}_{j-1}, \mathbf{q}_j, \mathbf{q}_{j+1})$ y $(\mathbf{q}_{k-1}, \mathbf{q}_k, \mathbf{p})$ respectivamente. Se calcula el signo de un triángulo formado por los vértices de siguiente forma: si el punto \mathbf{q}_j está a la derecha de la línea que une los vértices \mathbf{q}_{j-1} y \mathbf{q}_{j+1} el signo del área es positivo, si se encuentra a la izquierda el signo es negativo.

De forma simplificada, la expresión 3.7 quiere decir que la constante correspondiente a cada vértice, puede obtenerse como el producto del área del triángulo formado por el vértice actual y los dos vértices adyacentes a él y las áreas de los triángulos interiores formados por el punto \mathbf{p} y los demás vértices excluyendo las áreas que incluyan como vértice el punto \mathbf{q}_j .

Por ejemplo el peso w_j correspondiente al vértice \mathbf{q}_j en el polígono de 5 lados de la figura 3.6, se obtendría como: $w_j = A(\mathbf{q}_{j-1}, \mathbf{q}_j, \mathbf{q}_{j+1}) \cdot A_1 \cdot A_2 \cdot A_3$

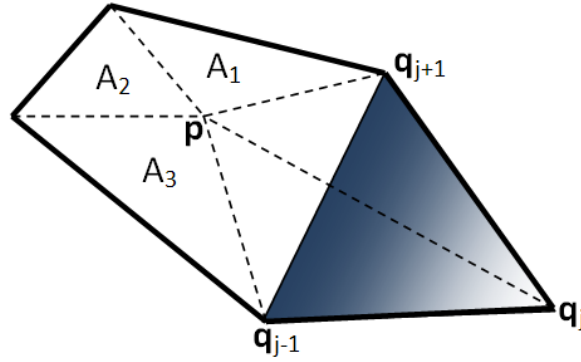


Figura 3.6: Coordenadas Baricéntricas Generalizadas. Obtención de la constante w_j

En el caso particular del triángulo de la figura 3.5, las constantes w_j obtenidas para cada vértice son:

$$\begin{aligned}
 w_1 &= A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) \cdot A_1 \\
 w_2 &= A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) \cdot A_2 \\
 w_3 &= A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) \cdot A_3 \\
 \sum w &= A^2(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)
 \end{aligned} \tag{3.8}$$

Donde A_1 , A_2 y A_3 son las áreas formadas por los triángulos con vértices en los puntos: $(\mathbf{q}_2, \mathbf{q}_3, \mathbf{p})$; $(\mathbf{q}_1, \mathbf{q}_3, \mathbf{p})$; $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{p})$ respectivamente.

Teniendo en cuenta todo lo anterior y aplicando la ecuación 3.6, el punto \mathbf{p} puede obtenerse como:

$$\begin{aligned}
 \mathbf{p} &= \frac{A_1 \cdot \mathbf{q}_1 + A_2 \cdot \mathbf{q}_2 + A_3 \cdot \mathbf{q}_3}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} = \frac{A_1 \cdot \mathbf{q}_1 + A_2 \cdot (\mathbf{q}_2 - \mathbf{q}_1 + \mathbf{q}_1) + A_3 \cdot (\mathbf{q}_3 - \mathbf{q}_1 + \mathbf{q}_1)}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \\
 &= \mathbf{q}_1 + \frac{A_2}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \mathbf{v}_1 + \frac{A_3}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \mathbf{v}_2 \Rightarrow \begin{cases} \mathbf{v}_1 = \mathbf{q}_2 - \mathbf{q}_1 \\ \mathbf{v}_2 = \mathbf{q}_3 - \mathbf{q}_1 \end{cases}
 \end{aligned} \tag{3.9}$$

El siguiente paso consiste en calcular los valores de las áreas A_1 , A_2 y A_3 . Éstas pueden obtenerse multiplicando el área del triángulo coincidente en \mathbf{m} por el coseno del ángulo existente entre ambos triángulos. Por ejemplo el área A_2 se puede calcular como el área del triángulo formado por los vértices $(\mathbf{q}_1, \mathbf{q}_3, \mathbf{m})$ multiplicada por el coseno del ángulo ϕ_{24} , dando lugar a la expresión:

$$\frac{A_2}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} = \frac{A(\mathbf{q}_1, \mathbf{q}_3, \mathbf{m})}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \cos(\phi_{24}) = \frac{-\sqrt{D(\mathbf{q}_1, \mathbf{q}_3, \mathbf{m})}}{\sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}} \cos(\phi_{24}) \tag{3.10}$$

Donde el signo '-' en la primera ecuación se debe al signo del área A_2 . De igual manera se calcula el área A_3 como:

$$\frac{A_3}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} = \frac{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{m})}{A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \cos(\phi_{34}) = \frac{\sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{m})}}{\sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}} \cos(\phi_{34}) \quad (3.11)$$

Aplicando las propiedades de los determinantes de Cayley-Menger (apartado 3.1.1) se obtiene que:

$$\begin{aligned} \cos(\phi_{24}) &= \frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_3, \mathbf{m})}{\sqrt{D(\mathbf{q}_1, \mathbf{q}_3, \mathbf{m})} \sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}} \\ \cos(\phi_{34}) &= \frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{m})}{\sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{m})} \sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}} \end{aligned} \quad (3.12)$$

Sustituyendo 3.10 y 3.12 en 3.6 se puede obtener \mathbf{p} como:

$$\mathbf{p} = \mathbf{q}_1 - \frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_3, \mathbf{m})}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \mathbf{v}_1 + \frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{m})}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \mathbf{v}_2 \quad (3.13)$$

Una vez obtenido el punto \mathbf{p} según [TR05] se puede calcular \mathbf{m} como:

$$\mathbf{m} = \mathbf{p} \pm \frac{h}{\|\mathbf{v}_1 \times \mathbf{v}_2\|} (\mathbf{v}_1 \times \mathbf{v}_2) \quad (3.14)$$

Donde h es la altura del tetraedro. El volumen de un tetraedro $V(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{m})$ puede obtenerse como $\frac{1}{3} \cdot A_{base} \cdot h$, por lo que utilizando las propiedades del apartado 3.1.1 se puede calcular la altura como:

$$h = 3 \cdot V(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{m}) / A(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) = \sqrt{\frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{m})}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)}} \quad (3.15)$$

Además:

$$\|\mathbf{v}_1 \times \mathbf{v}_2\| = \sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \quad (3.16)$$

Para simplificar la notación, se define $\mathbf{v}_3 = (\mathbf{v}_1 \times \mathbf{v}_2)$, por lo que el punto \mathbf{m} se puede obtener como:

$$\begin{aligned} \mathbf{m} &= \mathbf{q}_1 + \frac{1}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} (-D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_3, \mathbf{m}) \cdot \mathbf{v}_1 \\ &\quad + D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{m}) \cdot \mathbf{v}_2 \pm \left(\sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{m})} \right) \cdot \mathbf{v}_3) \end{aligned} \quad (3.17)$$

De la expresión anterior se pueden extraer varias conclusiones, la primera es que si los tres puntos conocidos estuviesen alineados el valor de $D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ es cero, por lo que aparecería una singularidad y sería imposible encontrar la posición de \mathbf{m} . Esto también indica que cuanto mayor sea el área entre los tres puntos, menor será el error cometido en la localización del punto para un mismo nivel de ruido en las medidas. La siguiente conclusión es que este método proporciona directamente la posición del punto a localizar en función de las medidas realizadas, por lo que no es necesario la utilización de métodos iterativos para obtener la ubicación del punto a localizar.

3.3. Posicionamiento hiperbólico

Uno de los problemas de la trilateración esférica consiste en que en el nodo a localizar se necesitaría conocer el instante de emisión de las balizas, por lo que debe existir algún método de sincronismo entre él mismo y las balizas. En la trilateración hiperbólica no es necesario conocer los instantes de emisión de las balizas, sino la diferencia de tiempos entre la recepción de la señal respecto de una baliza de referencia y las demás. En esta sección se van a desarrollar diferentes algoritmos de posicionamiento hiperbólicos basados en los determinantes de Cayley-Menger. Estos algoritmos obtienen la distancia a la primera baliza y una vez obtenida dicha distancia se podrá calcular la posición del nodo móvil.

En una primera aproximación se adapta la ecuación obtenida en el caso esférico, (ecuación 3.17) al caso de trilateración hiperbólica, esto es, se obtiene una expresión en la que la posición del nodo móvil sólo sea función de la distancia a la baliza de referencia y las diferencias de distancias que existen entre cada una de las balizas y la baliza de referencia, ya que esta diferencia es la que se puede medir en el caso hiperbólico. A continuación se proponen diferentes métodos que obtienen la distancia a la baliza de referencia, permitiendo de esa forma calcular la posición del nodo móvil.

En la figura 3.7 se muestra una representación de la adaptación de posicionamiento esférico a hiperbólico. En lo que sigue se supondrá, sin pérdida de generalidad, que la baliza de referencia es la primera baliza, \mathbf{p}_1 .

Suponiendo que \mathbf{p}_1 sea la baliza de referencia (figura 3.7) la distancia del resto de balizas al nodo móvil en función de la distancia a la baliza de referencia, l , puede expresarse como:

$$d(\mathbf{p}_j, \mathbf{m}) = d(\mathbf{p}_1, \mathbf{m}) + \Delta d(\mathbf{p}_j, \mathbf{p}_1) = l + \Delta d(\mathbf{p}_j, \mathbf{p}_1) \text{ con } j \neq 1 \quad (3.18)$$

Siendo l la distancia del nodo móvil a la primera baliza e $\Delta d(\mathbf{p}_j, \mathbf{p}_1)$ la diferencia de distancia existente entre la baliza j y la de referencia. Esta distancia es la que se puede medir durante la localización.

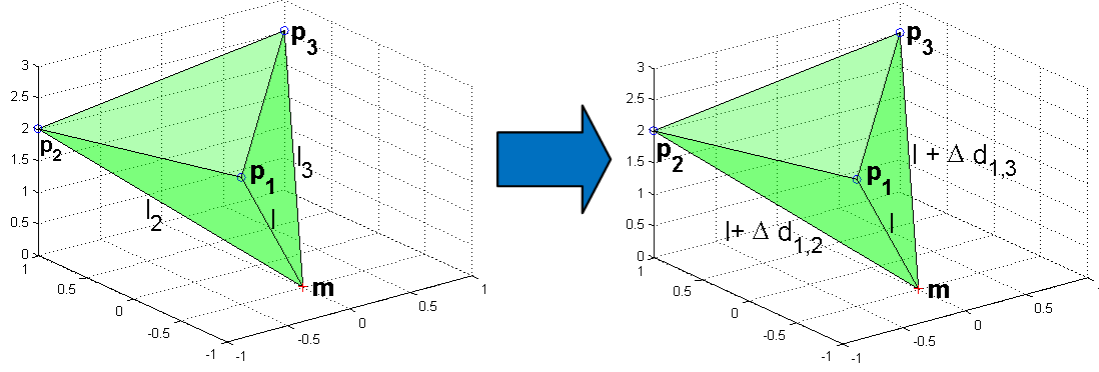


Figura 3.7: Paso de trilateración esférica a hiperbólica

Sustituyendo la expresión 3.18 en la ecuación 3.17 y desarrollando los determinantes de Cayley- Menger se llega a la siguiente expresión:

$$\mathbf{m} = \mathbf{p}_1 + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} [(b_1 \cdot l + c_1) \mathbf{v}_1 + (b_2 \cdot l + c_2) \mathbf{v}_2 \pm \left(\sqrt{a_3 \cdot l^2 + b_3 \cdot l + c_3} \right) \cdot \mathbf{v}_3] \quad (3.19)$$

Siendo:

$$\begin{aligned} b_1 &= \frac{\Delta d(\mathbf{p}_3, \mathbf{p}_1)}{2} [D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_3)] - D(\mathbf{p}_1, \mathbf{p}_3) \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) \\ b_2 &= \frac{\Delta d(\mathbf{p}_2, \mathbf{p}_1)}{2} [D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] + D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1) \\ c_1 &= \frac{1}{4} \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \cdot [D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2)] \\ &\quad + \frac{1}{4} \cdot D(\mathbf{p}_1, \mathbf{p}_3) \cdot [D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3) - 2\Delta d(\mathbf{p}_2, \mathbf{p}_1)^2] \\ c_2 &= \frac{1}{4} \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot [D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] \\ &\quad + \frac{1}{4} \cdot D(\mathbf{p}_1, \mathbf{p}_2) \cdot [D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_3) - 2\Delta d(\mathbf{p}_3, \mathbf{p}_1)^2] \\ a_3 &= [D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_2, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1) \\ &\quad - \frac{1}{4} \left\{ [D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2)]^2 + D(\mathbf{p}_2, \mathbf{p}_3)^2 - 2D(\mathbf{p}_1, \mathbf{p}_2) - 2D(\mathbf{p}_1, \mathbf{p}_3) \right\} \\ &\quad - \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \end{aligned} \quad (3.20)$$

$$\begin{aligned}
b_3 &= \frac{1}{4} \cdot \left\{ [D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_3) \right. \\
&\quad + [D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_3) \\
&\quad + [D(\mathbf{p}_1, \mathbf{p}_3) + D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2)] \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_2) \Big\} \\
&\quad - \frac{1}{4} \cdot \left\{ D(\mathbf{p}_1, \mathbf{p}_3) \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^4 + D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^4 \right. \\
&\quad + [D(\mathbf{p}_{12}, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \\
&\quad \left. + D(\mathbf{p}_1, \mathbf{p}_2) \cdot D(\mathbf{p}_1, \mathbf{p}_3) \cdot D(\mathbf{p}_2, \mathbf{p}_3) \right\} \\
c_3 &= -D(\mathbf{p}_1, \mathbf{p}_3) \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) - D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^3 \\
&\quad + \frac{1}{2} \cdot \left\{ [D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) \cdot D(\mathbf{p}_1, \mathbf{p}_3) \right. \\
&\quad + [-D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1) \cdot D(\mathbf{p}_1, \mathbf{p}_2) \Big\} \\
&\quad + \frac{1}{2} \cdot \left\{ \left[\Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \Delta d(\mathbf{p}_3, \mathbf{p}_1) + \Delta d(\mathbf{p}_2, \mathbf{p}_1) \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \right] \right. \\
&\quad \left. + [D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_3)] \right\}
\end{aligned}$$

Como se observa en la ecuación 3.19 la posición del punto \mathbf{m} queda sólo en función de la distancia a la primera baliza $\mathbf{m} = f(l)$, ya que las constantes $b_1, c_1, b_2, c_2, a_3, b_3, c_3$ son conocidas, pues sólo dependen de los incrementos de distancia medidos y de las distancias entre las balizas.

Los algoritmos propuestos obtienen la distancia a la primera baliza l siempre que el LPS esté constituido por al menos 4 balizas, lo cual permite crear más de una tripleta y poder de esa forma plantear un sistema de ecuaciones. Si el sistema estuviera compuesto por L balizas se podrían formar $\binom{L}{2}$ tripletas diferentes que contuvieran la baliza de referencia, pero de todas estas posibles tripletas sólo aportan nueva información aquellas tripletas que contengan al menos una baliza que no haya sido empleada por tripletas anteriores, por lo que sólo hay disponible $L-2$ tripletas.

Una vez calculada esta distancia, se obtiene la posición del nodo móvil resolviendo la ecuación 3.19. En las siguientes secciones se presentan tres métodos que solucionan el problema de la localización desde perspectivas diferentes.

3.3.1. Método 1: Localización con balizas en Z constante (LZC)

Como se ha mencionado, para obtener la posición del nodo móvil hay que solucionar una ecuación no lineal (ecuación 3.19). Se puede observar que el término que introduce la no linealidad es $\left(\sqrt{a_3 \cdot l^2 + b_3 \cdot l + c_3} \right) \cdot \mathbf{v}_3$, por lo que si se consiguiese anular este término el sistema que quedaría sería lineal. Si se supone que todas las balizas se encuentran a la

misma altura, esto es en un plano Z constante, los vectores \mathbf{v}_1 y \mathbf{v}_2 quedan de la forma $(v_{x1}, v_{y1}, 0)$ y $(v_{x2}, v_{y2}, 0)$ por lo que $\mathbf{v}_3 = (\mathbf{v}_1 \times \mathbf{v}_2)$ queda cómo $(0, 0, v_{z3})$. Si se expande la ecuación 3.19 en cada una de sus componentes se llega a:

$$\begin{cases} m_x = p_{x1} + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} [(b_1 \cdot l + c_1) \cdot v_{x1} + (b_2 \cdot l + c_2) \cdot v_{x2}] \\ m_y = p_{y1} + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} [(b_1 \cdot l + c_1) \cdot v_{y1} + (b_2 \cdot l + c_2) \cdot v_{y2}] \\ m_z = p_{z1} \pm \left(\sqrt{a_3 \cdot l^2 + b_3 \cdot l + c_3} \right) \cdot v_{z3} \end{cases} \quad (3.21)$$

Consiguiendo de esta forma que las componentes m_x y m_y sean ahora lineales, mientras que la componente m_z queda de forma cuadrática. Si el sistema LPS tiene una cuarta baliza que cumpla con la condición de la altura, se podría formar una nueva tripleta que permite plantear otro sistema de ecuaciones consiguiéndose de esta forma poder obtener l . Cabe destacar que todas las nuevas tripletas que se utilicen deben contener el punto \mathbf{p}_1 ya que es la distancia l el valor que se quiere encontrar.

En la figura 3.8 se puede ver que la posición de $\mathbf{m} = (m_x^{(1)}, m_y^{(1)}, m_z^{(1)})$ obtenida utilizando la primera tripleta de balizas $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ debe ser la misma, $\mathbf{m} = (m_x^{(2)}, m_y^{(2)}, m_z^{(2)})$, que la obtenida con la tripleta $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)$. En lo que sigue el superíndice 1 se referirá a los parámetros obtenidos con la primera tripleta, y el superíndice 2 a los obtenidos con la segunda tripleta.

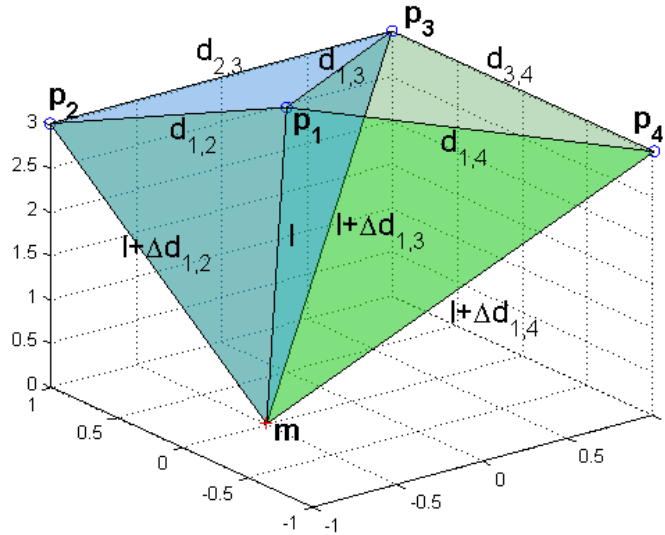


Figura 3.8: Esquema de posicionamiento para el método LZC.

$$\mathbf{m}^{(1)} = \mathbf{m}^{(2)} \Rightarrow \begin{pmatrix} m_x^{(1)} \\ m_y^{(1)} \\ m_z^{(1)} \end{pmatrix} = \begin{pmatrix} m_x^{(2)} \\ m_y^{(2)} \\ m_z^{(2)} \end{pmatrix}$$

$$\left\{ \begin{aligned} \frac{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[\left(b_1^{(1)} \cdot l + c_1^{(1)} \right) \cdot v_{x1}^{(1)} + \left(b_2^{(1)} \cdot l + c_2^{(1)} \right) \cdot v_{x2}^{(1)} \right] &= \left[\left(b_1^{(2)} \cdot l + c_1^{(2)} \right) \cdot v_{x1}^{(2)} \right. \\ &\quad \left. + \left(b_2^{(2)} \cdot l + c_2^{(2)} \right) \cdot v_{x2}^{(2)} \right] \\ \frac{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[\left(b_1^{(1)} \cdot l + c_1^{(1)} \right) \cdot v_{y1}^{(1)} + \left(b_2^{(1)} \cdot l + c_2^{(1)} \right) \cdot v_{y2}^{(1)} \right] &= \left[\left(b_1^{(2)} \cdot l + c_1^{(2)} \right) \cdot v_{y1}^{(2)} \right. \\ &\quad \left. + \left(b_2^{(2)} \cdot l + c_2^{(2)} \right) \cdot v_{y2}^{(2)} \right] \\ \left(\sqrt{a_3^{(1)} \cdot l^2 + b_3^{(1)} \cdot l + c_3^{(1)}} \right) \cdot \mathbf{v}_{z3}^{(1)} &= \left(\sqrt{a_3^{(2)} \cdot l^2 + b_3^{(2)} \cdot l + c_3^{(2)}} \right) \cdot \mathbf{v}_{z3}^{(2)} \end{aligned} \right. \quad (3.22)$$

Este método propuesto, como se ha mencionado, sólo es válido bajo la restricción de que todas las balizas se encuentren a la misma altura. A priori, esta suposición parece muy estricta, pero en la práctica no lo es tanto ya que normalmente las balizas se suelen ubicar en los techos de las habitaciones, consiguiéndose de esta forma que se cumpla la restricción impuesta.

3.3.2. Método 2: Localización por tripletas rotadas (LTR)

En la sección anterior se ha comprobado que si las balizas se encuentran a diferente altura, la ecuación a resolver es no lineal, (ecuación 3.19). El método propuesto en esta sección resuelve el sistema de ecuaciones rotando el eje de referencia para que al menos tres de las balizas sí se encuentren en un plano Z constante. De esta forma se consigue que el sistema a resolver sea cuadrático, pudiéndose de esa manera obtener la distancia a la primera baliza de una forma directa. En la figura 3.9 se muestra gráficamente un esquema del método propuesto.

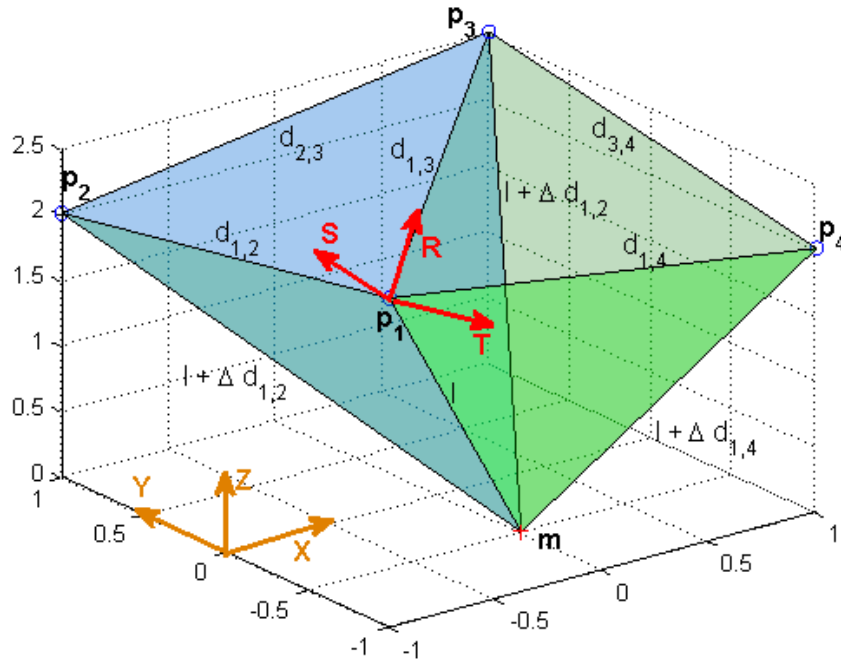


Figura 3.9: Esquema de posicionamiento rotando el sistema de referencia (LTR).

El primer paso del algoritmo consiste en seleccionar una matriz de rotación \mathbf{R} tal que, en el nuevo sistema propuesto los vectores $\mathbf{v}_1^{(1)}$, $\mathbf{v}_2^{(1)}$, $\mathbf{v}_3^{(1)}$ de la primera tripleta queden en el nuevo sistema rotado en el mismo plano Z constante:

$$\begin{aligned} \mathbf{v}_1^{(1R)} &= (v_{x1}^{(1R)}, 0, 0) \\ \mathbf{v}_2^{(1R)} &= (v_{x2}^{(1R)}, v_{y2}^{(1R)}, 0) \\ \mathbf{v}_3^{(1R)} &= (\mathbf{v}_1^{(1R)} \times \mathbf{v}_2^{(1R)}) = (0, 0, v_{z3}^{(1R)}) \end{aligned} \quad (3.23)$$

Donde $\mathbf{v}_1^{(1R)}$ y $\mathbf{v}_2^{(1R)}$ son los correspondientes vectores rotados de $\mathbf{v}_1^{(1)}$ y $\mathbf{v}_2^{(1)}$ de la primera tripleta $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$, el superíndice (1R indica que el sistema de referencia ha sido rotado y que los vectores se corresponden a la primera tripleta.

Existen diversos métodos para obtener una matriz de rotación que ubique tres balizas en el mismo plano. En el algoritmo implementado se ha seleccionado el que permite que los vectores \mathbf{v}_1 y \mathbf{v}_2 sean de la forma $(0, v_{y1}, 0)$ y $(v_{x2}, v_{y2}, 0)$ respectivamente, ya que de esta manera se pueden simplificar los cálculos.

El método para realizar la rotación es el siguiente, si $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3$ son los puntos que se desean que estén en el mismo plano (figura 3.9), entonces si se definen los vectores directores del nuevo sistema de referencia $(\vec{r}, \vec{s}, \vec{t})$ de la forma:

$$\vec{\mathbf{r}} = \frac{\mathbf{p}_2 - \mathbf{p}_1}{|\mathbf{p}_2 - \mathbf{p}_1|} \quad \vec{\mathbf{t}} = \frac{\vec{\mathbf{u}} \times (\mathbf{p}_3 - \mathbf{p}_1)}{|\vec{\mathbf{u}} \times (\mathbf{p}_3 - \mathbf{p}_1)|} \quad \vec{\mathbf{s}} = \vec{\mathbf{w}} \times \vec{\mathbf{u}} \quad (3.24)$$

Se cumple que la matriz de rotación buscada puede expresarse como:

$$\mathbf{R} = \begin{bmatrix} r_1 & s_1 & t_1 \\ r_2 & s_2 & t_2 \\ r_3 & s_3 & t_3 \end{bmatrix} \quad (3.25)$$

En el nuevo sistema de referencia se puede obtener la posición del punto \mathbf{m} como:

$$\begin{cases} m_x^{(1R)} = p_{x1}^{(R)} + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[(b_1^{(1)}l + c_1^{(1)})\mathbf{v}_{x1}^{(1R)} + (b_2^{(1)}l + c_2^{(1)})\mathbf{v}_{x2}^{(1R)} \right] \\ m_y^{(1R)} = p_{y1}^{(R)} + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[(b_2^{(1)}l + c_2^{(1)})\mathbf{v}_{y2}^{(1R)} \right] \\ m_z^{(1R)} \end{cases} \quad (3.26)$$

En el sistema de ecuaciones 3.26 no se desarrolla la expresión de la componente $m_z^{(1R)}$ porque daría lugar a una ecuación no lineal de orden superior al cuadrático y no es necesaria para solucionar el sistema. También cabe destacar que aunque se ha rotado el sistema de referencia, las constantes $b_1^{(1)}, b_2^{(1)}, c_1^{(1)}, c_2^{(1)}, a_3^{(1)}, b_3^{(1)}, c_3^{(1)}$ no varían su valor ya que sólo dependen de las distancias entre balizas y las diferencias de distancias medidas y estos valores no cambian aunque se aplique una rotación al sistema de referencia.

Para la segunda tripleta $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)$ se obtienen los siguientes vectores rotados:

$$\begin{aligned} \mathbf{v}_1^{(2R)} &= \left(v_{x1}^{(2R)}, v_{y1}^{(2R)}, v_{z1}^{(2R)} \right) \\ \mathbf{v}_2^{(2R)} &= \left(v_{x2}^{(2R)}, v_{y2}^{(2R)}, v_{z2}^{(2R)} \right) \\ \mathbf{v}_3^{(2R)} &= \left(\mathbf{v}_1^{(2R)} \times \mathbf{v}_2^{(2R)} \right) = \left(v_{x3}^{(2R)}, v_{y3}^{(2R)}, v_{z3}^{(2R)} \right) \end{aligned} \quad (3.27)$$

Igual que en el caso anterior el superíndice (2R indica que el sistema de referencia ha sido rotado, y que es la segunda tripleta. En este caso como se emplea la misma matriz de rotación que en la primera tripleta, no tiene por qué hacerse cero ninguna de las componentes de los vectores $\mathbf{v}_1^{(2R)}, \mathbf{v}_2^{(2R)}, \mathbf{v}_3^{(2R)}$.

La posición de las componentes x e y de \mathbf{m} usando esta tripleta pueden calcularse como:

$$\left\{ \begin{array}{l} m_x^{(2R)} = p_{x1}^{(R)} + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)} \left[\left(b_1^{(2)}l + c_1^{(2)} \right) \mathbf{v}_{x1}^{(2R)} \right. \\ \quad \left. + \left(b_2^{(2)}l + c_2^{(2)} \right) \mathbf{v}_{x2}^{(1R)} \pm \left(\sqrt{a_3^{(2)}l^2 + b_3^{(2)}l + c_3^{(2)}} \right) \cdot \mathbf{v}_{x3}^{(2R)} \right] \\ \\ m_y^{(2R)} = p_{y1}^{(R)} + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)} \left[\left(b_1^{(2)}l + c_1^{(2)} \right) \cdot \mathbf{v}_{y1}^{(2R)} \right. \\ \quad \left. + \left(b_2^{(2)}l + c_2^{(2)} \right) \mathbf{v}_{y2}^{(1R)} \pm \left(\sqrt{a_3^{(2)}l^2 + b_3^{(2)}l + c_3^{(2)}} \right) \cdot \mathbf{v}_{y3}^{(2R)} \right] \end{array} \right. \quad (3.28)$$

A pesar de que se ha rotado el sistema de referencia, el punto a localizar debe ser el mismo para ambas tripletas $\mathbf{m}^{(1R)} = \mathbf{m}^{(2R)}$ por lo que igualando las ecuaciones 3.26 y 3.28 se obtiene el siguiente sistema de ecuaciones:

$$\left\{ \begin{array}{l} \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left(\left(b_1^{(1)}l + c_1^{(1)} \right) \mathbf{v}_{x1}^{(1R)} + \left(b_2^{(1)}l + c_2^{(1)} \right) \mathbf{v}_{x2}^{(1R)} \right) = \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)} \left[\left(b_1^{(2)}l + c_1^{(2)} \right) \mathbf{v}_{x1}^{(2R)} \right. \\ \quad \left. + \left(b_2^{(2)}l + c_2^{(2)} \right) \mathbf{v}_{x2}^{(2R)} \pm \left(\sqrt{a_3^{(2)}l^2 + b_3^{(2)}l + c_3^{(2)}} \right) \cdot \mathbf{v}_{x3}^{(2R)} \right] \\ \\ \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left(\left(b_2^{(1)}l + c_2^{(1)} \right) \mathbf{v}_{y2}^{(1R)} \right) = \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)} \left[\left(b_1^{(2)}l + c_1^{(2)} \right) \mathbf{v}_{y1}^{(2R)} \right. \\ \quad \left. + \left(b_2^{(2)}l + c_2^{(2)} \right) \mathbf{v}_{y2}^{(2R)} \pm \left(\sqrt{a_3^{(2)}l^2 + b_3^{(2)}l + c_3^{(2)}} \right) \cdot \mathbf{v}_{y3}^{(2R)} \right] \end{array} \right. \quad (3.29)$$

Por lo que se llega a un sistema de dos ecuaciones cuadráticas del tipo $a \cdot l^2 + b \cdot l + c = 0$ con una sola incógnita, la distancia a la primera baliza (l). Al resolver el sistema se obtienen dos posibles soluciones de l , para discriminar cuál de ellas es la adecuada se calcula la coordenada z de la baliza con ambas soluciones:

$$m_z^{(1R)} = \left(\sqrt{a_3^{(2)}l^2 + b_3^{(2)}l + c_3^{(2)}} \right) \cdot \mathbf{v}_{z3}^{(1R)} \quad (3.30)$$

Se ha comprobado que sólo se obtiene una coordenada z coherente para una de las soluciones de l .

Este proceso se repite con todas las posibles tripletas válidas en el entorno que contengan al punto p_1 para de esa forma conseguir sobredimensionar aún más el sistema. Una vez resuelto y obtenido el valor de l se puede obtener la posición de m empleando la ecuación 3.19 como en el caso anterior.

3.3.3. Método 3: Localización por tripletas con cálculo iterativo (LTI)

En este apartado se desarrolla un algoritmo válido tanto si las balizas están en el mismo plano Z constante, como si no lo están, además también permite integrar mejor el número de balizas.

Como se ha mencionado, para obtener el valor de l hay que resolver un sistema de ecuaciones no lineal; en los algoritmos que se han propuesto anteriormente se trataba de buscar alguna restricción geométrica para resolver de forma lineal el sistema de ecuaciones.

En el método que se presenta a continuación, se soluciona el sistema empleando un método de resolución para sistemas no lineales; en concreto se emplea el algoritmo de Gauss-Newton. Este algoritmo trata de encontrar el valor de l mediante un proceso iterativo: partiendo de un valor de l inicial (\hat{l}), se trata de encontrar el valor de l que consiga que la posición obtenida con cada tripla sea la misma (ver figura 3.10).

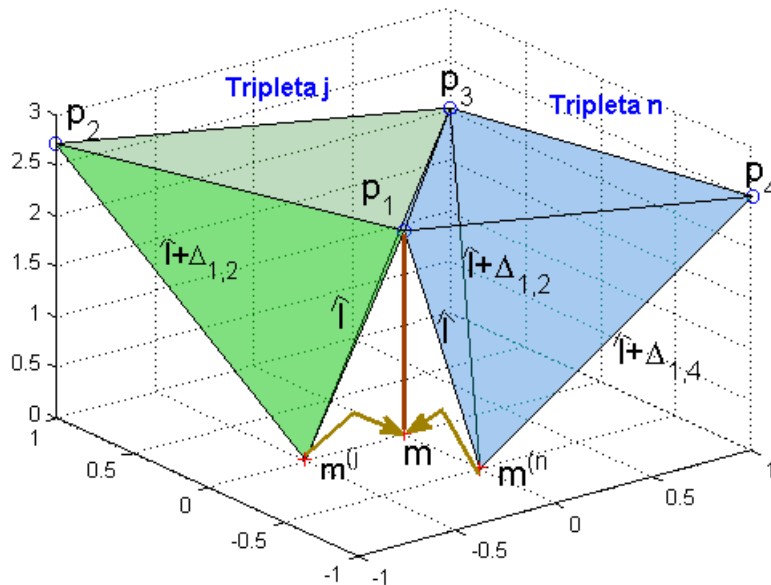


Figura 3.10: Esquema de posicionamiento para el método LTI.

Para ello se minimiza la función $F(\hat{l})$ que expresa el error cuadrático que existe entre las posiciones obtenidas con cada par de tripletas (j y n), si se emplea la distancia \hat{l} . Ya que para el valor correcto de \hat{l} , esto es cuando $\hat{l} = l$ todas las tripletas tendrán como solución el punto donde se encuentra el nodo móvil y el error será cero.

$$F(\hat{l}) = \sum_{j=1}^{T-1} \sum_{n=1}^T \sum_{i=x,y,z} \left(\mathbf{m}_i^{(j)} - \mathbf{m}_i^{(n)} \right)^2 = \sum_{j=1}^{T-1} \sum_{n=1}^T \sum_{i=x,y,z} [f_i(\hat{l})]^2 \quad (3.31)$$

Donde i indica la componente espacial (x, y, z) del nodo móvil.

De la ecuación 3.19 se obtiene que para cada tripleta j la posición obtenida puede expresarse como:

$$\begin{aligned} \mathbf{m}^{(j)} &= \mathbf{p}_1 + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[\left(b_1^{(j)} \hat{l} + c_1^{(j)} \right) \mathbf{v}_1^{(j)} + \left(b_2^{(j)} \hat{l} + c_2^{(j)} \right) \mathbf{v}_2^{(j)} \right. \\ &\quad \left. \pm \left(\sqrt{a_3^{(j)} \hat{l}^2 + b_3^{(j)} \hat{l} + c_3^{(j)}} \right) \cdot \mathbf{v}_3^{(j)} \right] \\ \mathbf{m}^{(n)} &= \mathbf{p}_1 + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[\left(b_1^{(n)} \hat{l} + c_1^{(n)} \right) \mathbf{v}_1^{(n)} + \left(b_2^{(n)} \hat{l} + c_2^{(n)} \right) \mathbf{v}_2^{(n)} \right. \\ &\quad \left. \pm \left(\sqrt{a_3^{(n)} \hat{l}^2 + b_3^{(n)} \hat{l} + c_3^{(n)}} \right) \cdot \mathbf{v}_3^{(n)} \right] \end{aligned} \quad (3.32)$$

Siendo \hat{l} la distancia estimada a la primera baliza. Una de las ventajas de usar esta fórmula es que el espacio de búsqueda de \hat{l} puede acotarse ya que sólo son válidos aquellos valores en los que $a_3^{(j)} \hat{l}^2 + b_3^{(j)} \hat{l} + c_3^{(j)}$ son positivos.

Derivando la ecuación 3.31 respecto a \hat{l} y reordenando el sistema de la forma $\mathbf{A} \cdot \Delta \mathbf{X} = \mathbf{B}$ se llega a:

$$\Delta \mathbf{X} = (\Delta l) \quad \mathbf{A} = \begin{pmatrix} \frac{\partial f_x}{\partial \hat{l}} \\ \frac{\partial f_y}{\partial \hat{l}} \\ \frac{\partial f_z}{\partial \hat{l}} \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} m_x^{(j)} - m_x^{(n)} \\ m_y^{(j)} - m_y^{(n)} \\ m_z^{(j)} - m_z^{(n)} \end{pmatrix} \quad (3.33)$$

Donde:

$$\begin{aligned} \frac{\partial f_i}{\partial \hat{l}} &= -a_1^{(j)} v_{i1}^{(j)} + a_2^{(j)} v_{i2}^{(j)} \pm \frac{\left(2c_3^{(j)} \hat{l} + a_3^{(j)} \right) v_{i3}^{(j)}}{2\sqrt{c_3^{(j)} \hat{l}^2 + a_3^{(j)} \hat{l} + b_3^{(j)}}} \\ &\quad + a_1^{(n)} v_{i1}^{(n)} - a_2^{(n)} v_{i2}^{(n)} \mp \frac{\left(2c_3^{(n)} \hat{l} + a_3^{(n)} \right) v_{i3}^{(n)}}{2\sqrt{c_3^{(n)} \hat{l}^2 + a_3^{(n)} \hat{l} + b_3^{(n)}}} \quad i = x, y, z \end{aligned} \quad (3.34)$$

Resolviendo el sistema por mínimos cuadrados:

$$\Delta \mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B} \quad (3.35)$$

Se obtiene el incremento (Δl) que se debe restar a la distancia \hat{l} anterior, por lo que el nuevo valor de \hat{l} se calcula de la forma $\hat{l}_k = \hat{l}_{k-1} - \Delta l$. Este proceso se repite hasta que los incrementos de \hat{l} , (Δl), sean inferiores al valor de un umbral predeterminado.

El principal problema del método consiste en que al realizar la derivación, (ecuación 3.34), en ciertos coeficientes aparece un signo \pm , por lo que es necesario averiguar el signo correcto para obtener la posición. Este signo depende de la configuración de las balizas y del punto a localizar.

Como se puede observar en la figura 3.11 el plano que se puede formar por la tripleta divide el espacio en dos zonas. Según en la zona en la que se encuentre el objeto móvil habrá que emplear un signo o el otro. Esto se refleja perfectamente cuando las balizas están en un plano Z constante, ya que en esta situación, el espacio por encima de las mismas emplea un signo y el espacio inferior a ellas emplea el signo contrario.

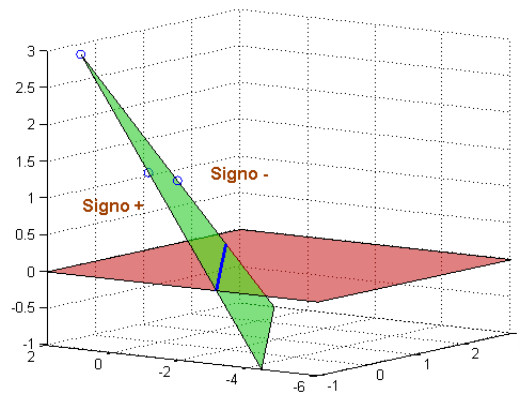


Figura 3.11: Elección del signo de la derivada.

En la mayoría de las situaciones las balizas se encuentran encima del móvil y la diferencia de altura no es tan grande como para que el plano formado por las balizas interseccione dentro del área de cobertura y haga necesario el cambio en la elección del signo. En esta situación para determinar que signo elegir hay fijarse en el signo de la componente 'z' del vector $\mathbf{v}_3 = (\mathbf{v}_1 \times \mathbf{v}_2)$: si esta componente es negativa hay que emplear el signo '+'; en caso contrario cuando la componente 'z' del vector \mathbf{v}_3 sea positiva, se debe emplear el signo '-'.

3.4. Resultados de simulaciones

Para demostrar las propiedades de los algoritmos de trilateración hiperbólica propuestos, se han realizado varias simulaciones comparando los resultados con la resolución directa de las ecuaciones de distancia mediante el algoritmo iterativo de Gauss-Newton (MGN, ver anexo A), ya que este método es de los más utilizados a la hora de realizar un posicionamiento hiperbólico [YTBJ08]. Para la realización de estas pruebas se emplea el

valor del indicador PDOP (*Dilución en la Precisión de la Posición*). Este parámetro, muy empleado en GPS [ZH06], permite evaluar la precisión del sistema. En el Apéndice B se hace una descripción más detallada del mismo.

Debido a que el método LZC sólo es válido cuando las balizas están en un plano Z constante se han estructurado las pruebas de la siguiente forma. En una primera parte se realizan pruebas para los métodos LZC y LTI con dos configuraciones diferentes de balizas; mientras que en una segunda parte se eligen otras dos configuraciones de balizas y se realizan pruebas para los métodos LTR y LTI

En todas las pruebas se supone un área de cobertura de $8 \times 8 \text{ m}^2$, y se realizan localizaciones en un grid de 17×17 puntos, ejecutándose en cada uno de esos puntos cada algoritmo de posicionamiento 580 veces, añadiéndole diferentes niveles de ruido Gaussiano a las medidas en cada localización. En total se habrán realizado 167.620 localizaciones por prueba. Los test se realizan en un PC Core 2 duo de 2.66 GHz (solo se emplea un procesador), con 3.5 Gb de RAM y como programa de cálculo se utiliza MATLAB 7.4

3.4.1. Pruebas para los métodos LZC y LTI

En la primera prueba se va a suponer un sistema LPS compuesto por 4 balizas que se encuentran situadas en el techo, esto es, todas a la misma altura, y bastante separadas unas de otras, consiguiendo de esta forma un PDOP bastante bueno [ML09]. En la figura 3.12 se muestra cómo queda distribuido el sistema LPS y la posición de las balizas.

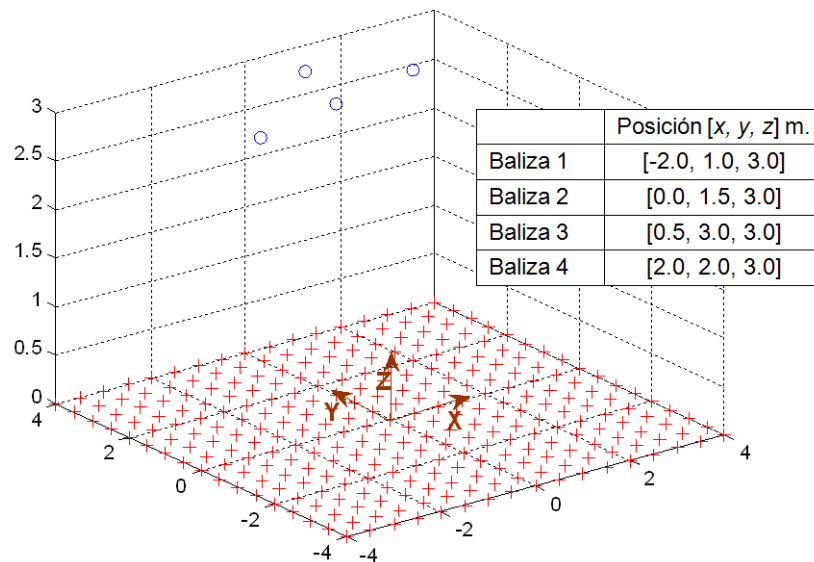


Figura 3.12: Distribución de las balizas y los puntos de test para la primera simulación.

En la figura 3.13 se muestra mediante curvas de nivel el PDOP obtenido con los diferentes métodos, se observa que en todos los métodos su valor oscila entre 10 y 60 y que la distribución es muy similar. En la tabla 3.1 se muestran las diferencias medias y máximas en porcentaje entre los valores del PDOP obtenidos con el método MGN hiperbólico

y los obtenidos con los métodos LZC (método 1) y LTI (método 3), se comprueba que los resultados obtenidos son muy semejantes ya que la diferencia media no sobrepasa el 1.25 %, de hecho, para el método LTI esta diferencia es menor del 1 %.

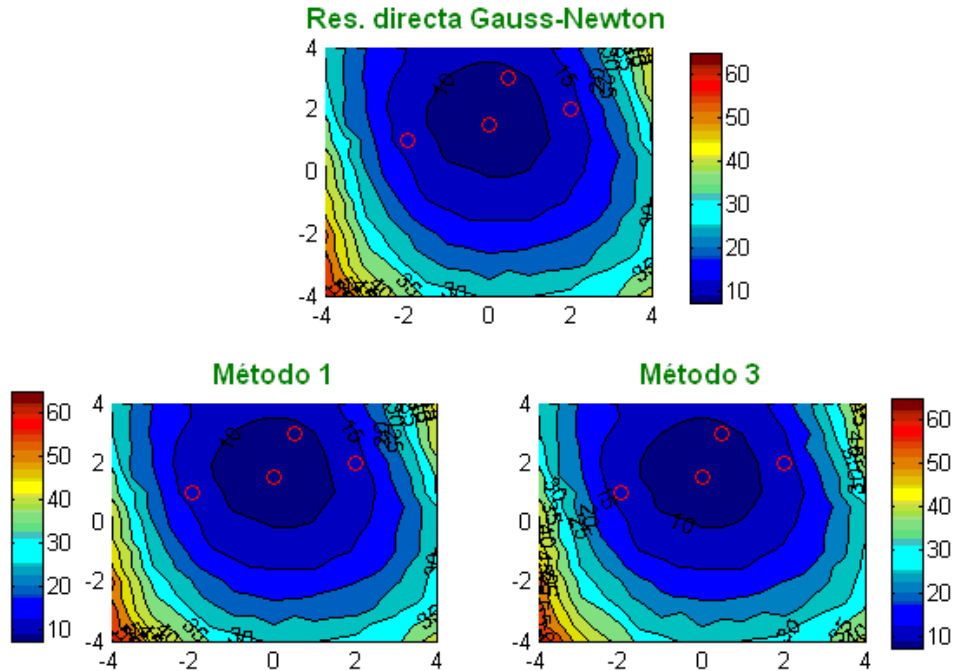


Figura 3.13: Comparativa PDOP obtenido entre los métodos propuestos y la resolución directa por Gauss-Newton, para la primera distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.

Tabla 3.1: Diferencias medias y máximas entre el algoritmo MGN con los algoritmos LZC y LTI en el primer test.

	Método LZC	Método LTI
Dif. Media	1.24 %	0.97 %
Dif. Máxima	3.70 %	4.52 %

Para poder comparar de forma analítica las semejanzas de los PDOP obtenidos, se ha calculado, para cada uno de los puntos donde se ha realizado la localización la diferencia en porcentaje entre el PDOP usando los métodos propuestos, y el PDOP empleando el método MGN hiperbólico. En la figura 3.14 se muestra el histograma de esta operación. Se puede comprobar que para ambos métodos en el 94 % de los 289 puntos de medida empleados la diferencia es inferior a un 2 %

La mayor diferencia entre los métodos, como se puede ver en la tabla 3.2, radica en la velocidad (carga computacional), ya que los algoritmos propuestos realizan el cálculo de la posición del nodo móvil de forma más rápida que la resolución directa empleando el método de Gauss-Newton. Esto se debe a diversas razones:

- El algoritmo LZY (método 1) es más rápido ya que es un algoritmo no iterativo.
- El método LTI (método 3) a pesar de ser un método iterativo, cumple una serie de requisitos que lo hacen ser más eficiente a efectos de tiempo de cómputo. El primero de ellos es que el espacio de búsqueda se limita a una dimensión (la distancia a la baliza de referencia), mientras que en el método de resolución directa empleando Gauss-Newton la búsqueda es tridimensional (las coordenadas (x,y,z) del objeto a localizar). Además el espacio de búsqueda en el caso del algoritmo basado en Cayley-Menger iterativo está más acotado ya que se conoce la distancia máxima y mínima entre las que están comprendidos los valores de la solución. Por último sólo son válidos valores de ' l ' que hagan que la raíz cuadrada de la ecuación 3.19 sea positiva, por lo que el espacio de búsqueda se puede acotar aún más.

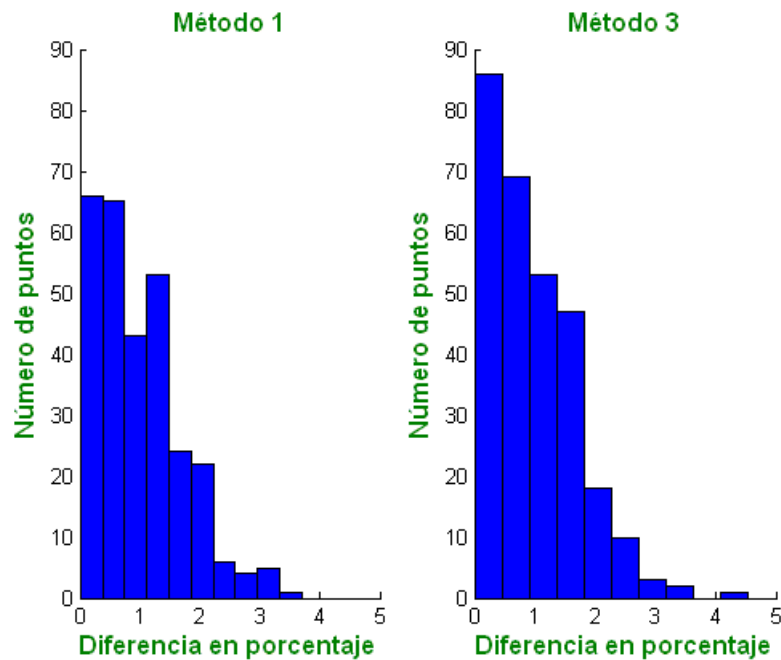


Figura 3.14: Histograma con el porcentaje en diferencia entre los métodos propuestos LZY y LTI y el método de resolución directa para la primera distribución de balizas. Izquierda, método LZY Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton.

Tabla 3.2: Tiempo necesario para finalizar los test en la primera simulación.

Res. Directa Gauss-Newton	Método LZY	Método LTI
21min. 15sg.	1min. 05sg	2min. 50sg.

En una segunda prueba se va a proponer una distribución de balizas mucho peor (véase la figura 3.15), esto es que la ubicación de las 4 balizas sea muy ineficiente y haga que el PDOP dentro del área de cobertura sea muy elevado. Las balizas siguen estando a la misma altura.

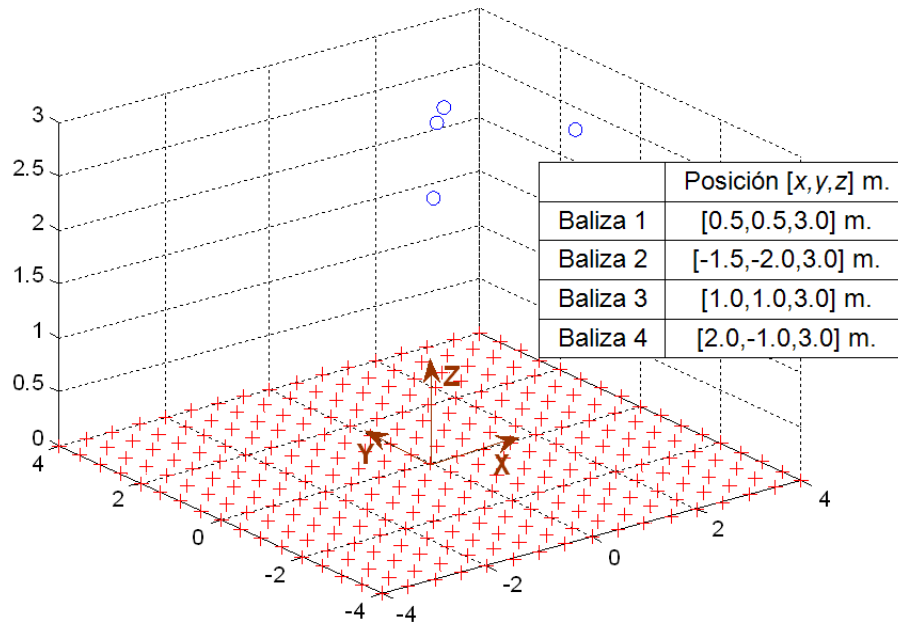


Figura 3.15: Distribución de las balizas y los puntos de test para la segunda simulación.

Se puede observar en la figura 3.16 que efectivamente el PDOP empleando esta distribución es mucho peor que en el caso anterior ya que varía entre 50 y 250. Además se puede observar visualmente que las gráficas son muy similares en los tres casos.

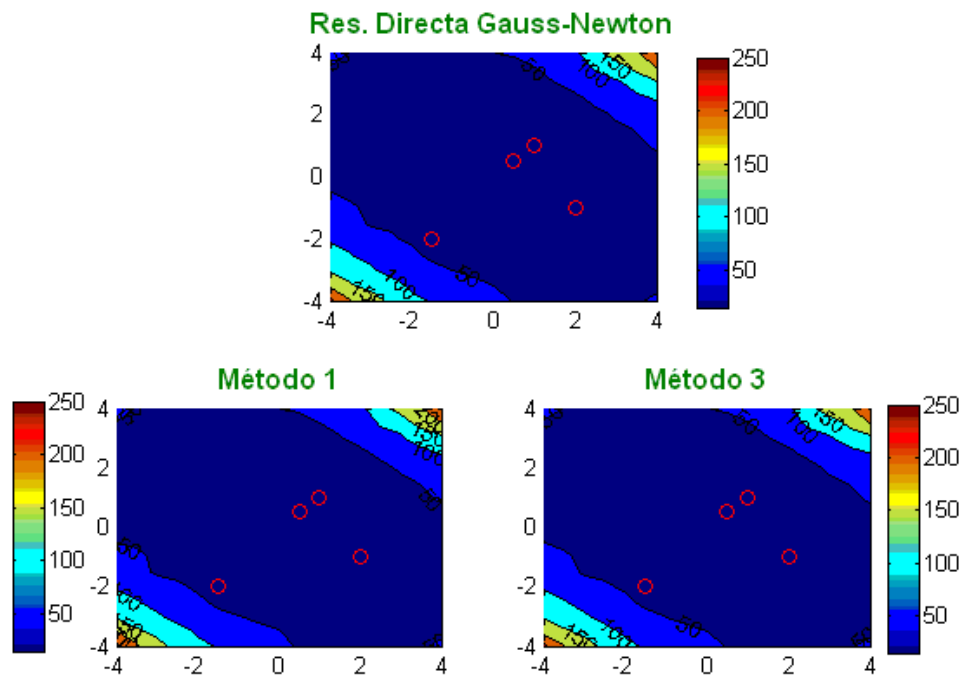


Figura 3.16: Comparativa PDOP obtenido entre los métodos propuestos y la resolución directa por Gauss-Newton, para la segunda distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.

En la tabla 3.3 se muestran las diferencias máximas y medias entre el algoritmo de MGN hiperbólico y los métodos LZC y LTI. Se puede comprobar que las diferencias medias siguen estando en torno al 1.25 %.

Tabla 3.3: Diferencias medias y máximas entre los algoritmos en el segundo test.

	Método LZC	Método LTI
Dif. Media	1.26 %	1.09 %
Dif. Máxima	5.51 %	4.01 %

En la figura 3.17 se puede observar un histograma con las diferencias expresadas en porcentaje de los diferentes métodos propuestos con el algoritmo de resolución directa de Gauss-Newton, se comprueba que para el 95 % de los casos la diferencia es menor al 2.5 % para ambos algoritmos.

Al igual que en el caso anterior los algoritmos propuestos demuestran ser más eficientes computacionalmente, que la resolución directa mediante Gauss-Newton (véase la tabla 3.4 con los tiempos de cómputo). Como la distribución de balizas es peor, el tiempo empleado para realizar las pruebas en el caso de los algoritmos no lineales es algo mayor que en el primer test.

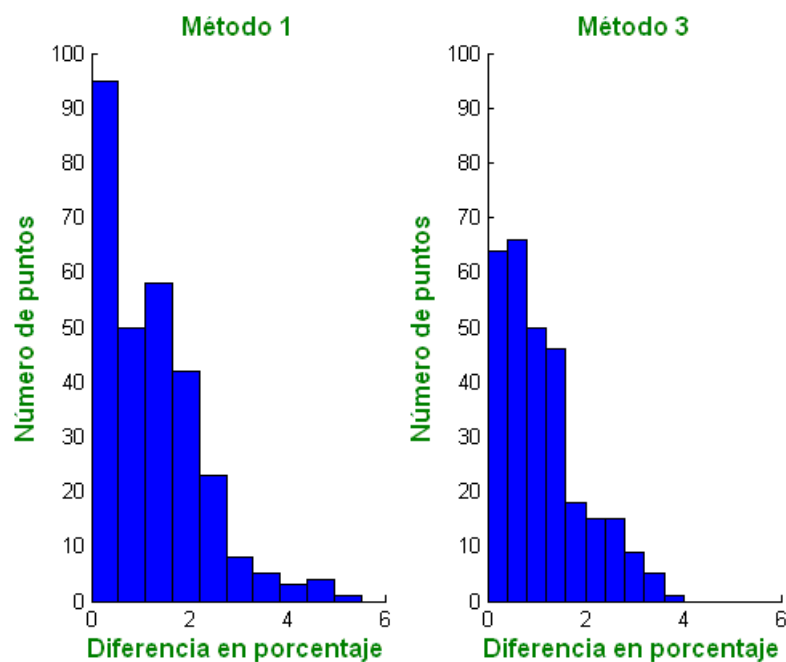


Figura 3.17: Histograma con el porcentaje en diferencia entre los métodos propuestos LZC y LTI y el método de resolución directa para la segunda distribución de balizas. Izquierda, método LZC Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton.

Tabla 3.4: Tiempo necesario para finalizar los test en la segunda simulación.

Gauss-Newton	Método LZC	Método LTI
23min. 51sg.	1min. 03sg	3min. 14sg.

3.4.2. Pruebas para los métodos LTR y LTI

En las siguientes simulaciones las balizas ya no se sitúan a la misma altura por lo cual ya no se puede emplear el método LZC pero se realizan pruebas con el método LTR (método 2) para comprobar sus propiedades. En las pruebas también se emplea el método LTI ya que sigue siendo válido en estas situaciones. Además se simula un sistema LPS compuesto por cinco balizas para comprobar el funcionamiento con más balizas en el entorno.

En esta tercera prueba se selecciona una distribución de balizas de tal forma que el PDOP dentro del área de cobertura sea muy bajo. De esta forma se comprueba el funcionamiento de los algoritmos en condiciones favorables, véase la figura 3.18.

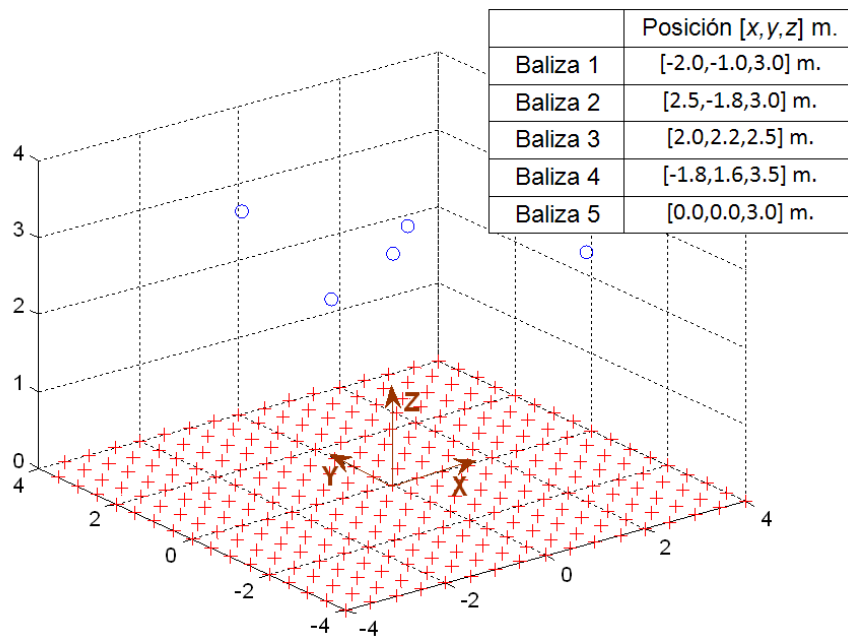


Figura 3.18: Distribución de las balizas y los puntos de test para la tercera prueba.

En la figura 3.19 se muestra el PDOP obtenido con los diferentes métodos: GN, LTR y LTI. Se puede observar que en el caso de emplear el método LTR el PDOP obtenido es peor que con los otros dos métodos, ya que en el método MGN y el algoritmo propuesto LTI varía entre 6 y 18 y en el caso del algoritmo de rotación del eje de referencia varía entre 10 y 30. Esto es porque debido a las rotaciones aparecen zonas donde se incrementa el error en la determinación de la posición del nodo móvil. Como la escala de los gráficos se ve alterada por el comportamiento del método LTR, en la figura 3.20 se muestran las

mismas gráficas solo que eliminando el PDOP del segundo método para comprobar que la distribución del PDOP es similar en ambos casos.

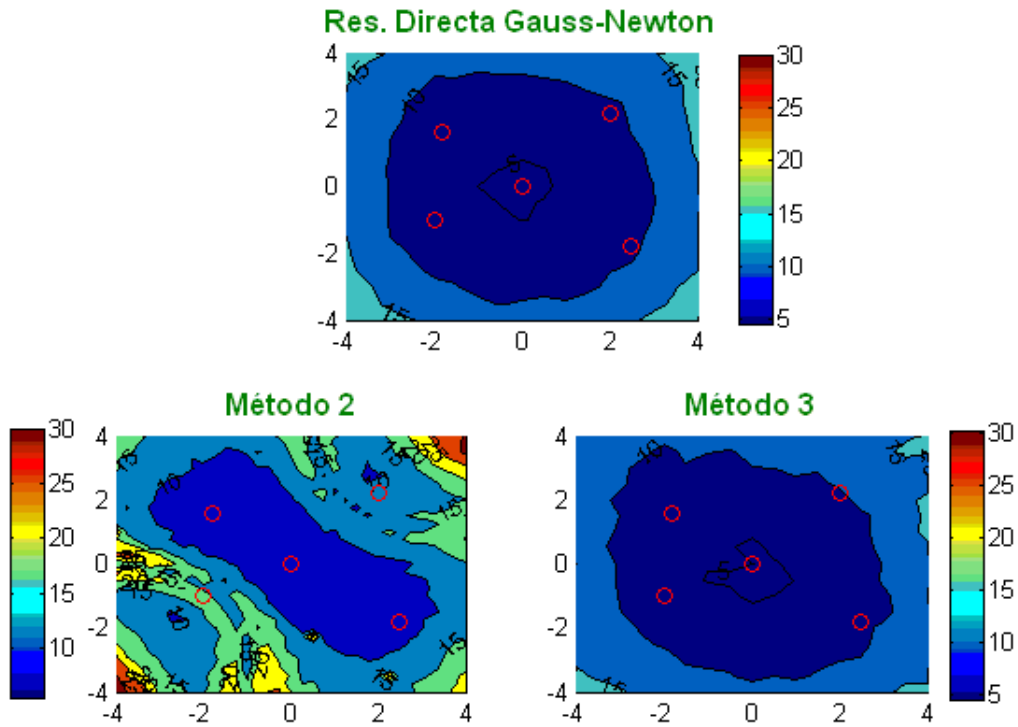


Figura 3.19: Comparativa PDOP obtenido entre los métodos propuestos LTR (método 2) y LTI (método 3) con el método de la resolución directa por Gauss-Newton, para la tercera distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.

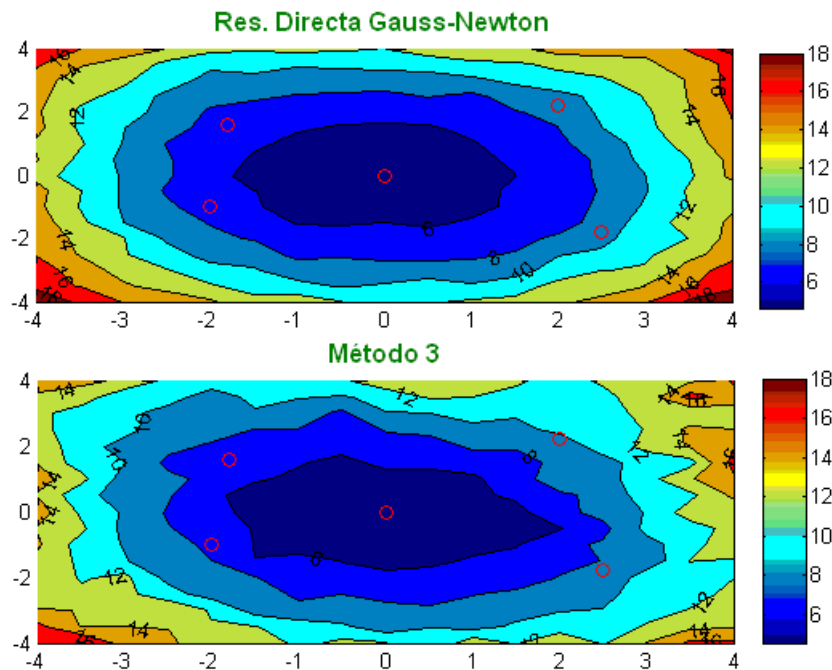


Figura 3.20: Comparativa PDOP obtenido entre el método LTI y la resolución directa mediante Gauss-Newton, para la tercera distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.

En la tabla 3.5 se muestran las diferencias medias y máximas entre los diferentes métodos, se puede comprobar que en el método LTR las diferencias son muy elevadas, la diferencia media es del 35.43 % y la máxima del 213 %, en el caso del método LTI las diferencias siguen estando en valores similares que en las otras pruebas.

Tabla 3.5: Diferencias medias y máximas entre los algoritmos en el tercer test.

	Método LTR	Método LTI
Dif. Media	35.43 %	1.04 %
Dif. Máxima	213 %	4.33 %

En la figura 3.21 se vuelve a mostrar un histograma con los valores en porcentaje de las diferencias obtenidas en los 289 puntos de test. Se puede comprobar que en el caso del método LTR las diferencias son muy elevadas. De hecho hay 18 puntos con un porcentaje superior al 100 %, esto es, que la diferencia es más que el doble que el PDOP del método directo de Gauss Newton, aunque el 80 % de los puntos tiene una diferencia inferior al 40 %. En cuanto al método LTI en el 90 % de los casos la diferencia es menor al 2 % del valor que en el caso de emplear el método de resolución directa de Gauss-Newton. En este caso como las escalas en el eje 'X' son muy diferentes no se han igualado.

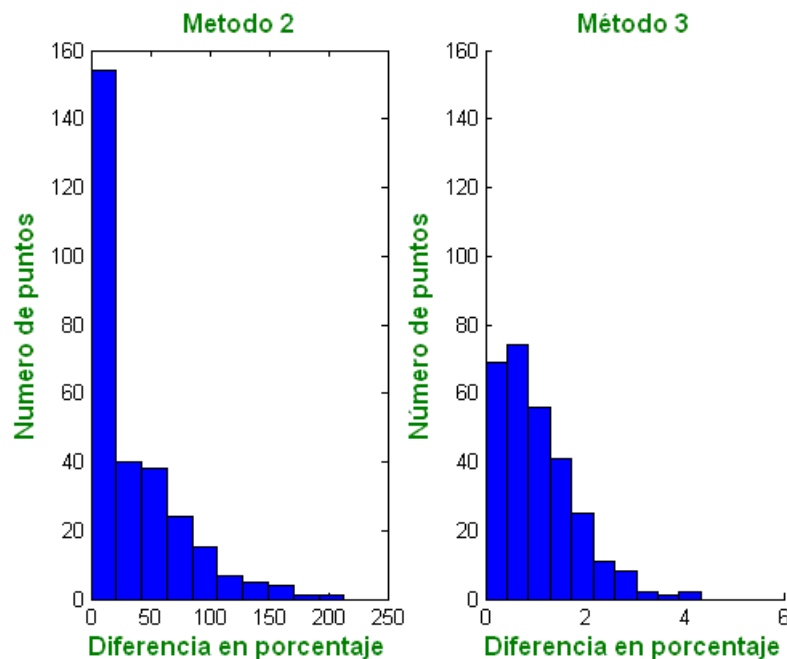


Figura 3.21: Histograma con el porcentaje en diferencia entre los métodos propuestos LTR y LTI y el método de resolución directa para la tercera distribución de balizas. Izquierda, método LTR Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton.

Al igual que en las demás pruebas se ha medido el tiempo necesario para su realización, véase Tabla 3.6. Se puede comprobar que los algoritmos propuestos son más eficientes en cuanto a tiempo de cálculo y que como el LPS está compuesto por cinco balizas estos tiempos se incrementan respecto a las pruebas anteriores.

Tabla 3.6: Tiempo necesario para finalizar los test en la tercera simulación.

Gauss-Newton	Método LTR	Método LTI
35min. 07sg.	2min. 36sg	5min. 12sg.

Por último se realiza una última prueba en la que las balizas se distribuyen a distinta altura pero de tal forma que el PDOP dentro del área de cobertura sea muy malo, así se evalúan los algoritmos en una situación de mala distribución de balizas, (véase la figura 3.22).

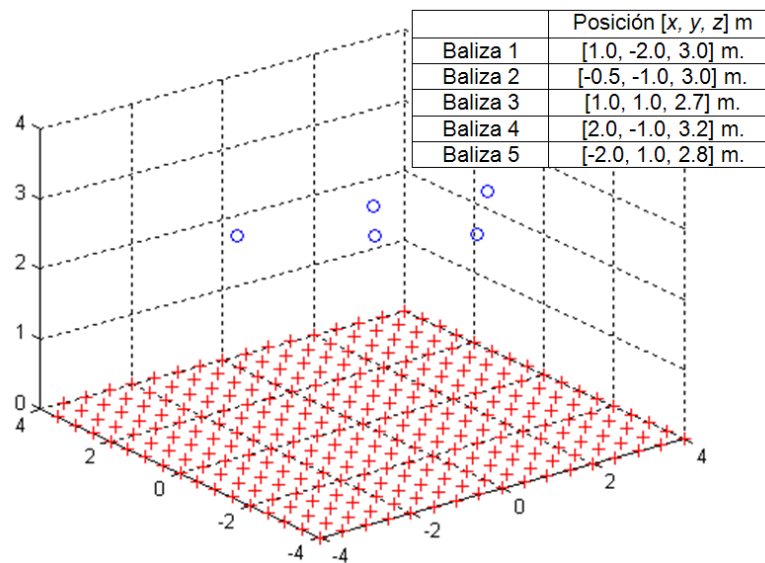


Figura 3.22: Distribución de las balizas y los puntos de test para la tercera prueba.

Como se puede observar en la figura 3.23 el PDOP con esta distribución de balizas es bastante peor que en los otros casos, ya que varía entre 10 y 200 en el caso del segundo método y entre 10 y 160 en los otros métodos. También se observa que la distribución de PDOP en el caso de emplear el algoritmo LTR esta vez si que es más similar que la del método de referencia que en la prueba anterior.

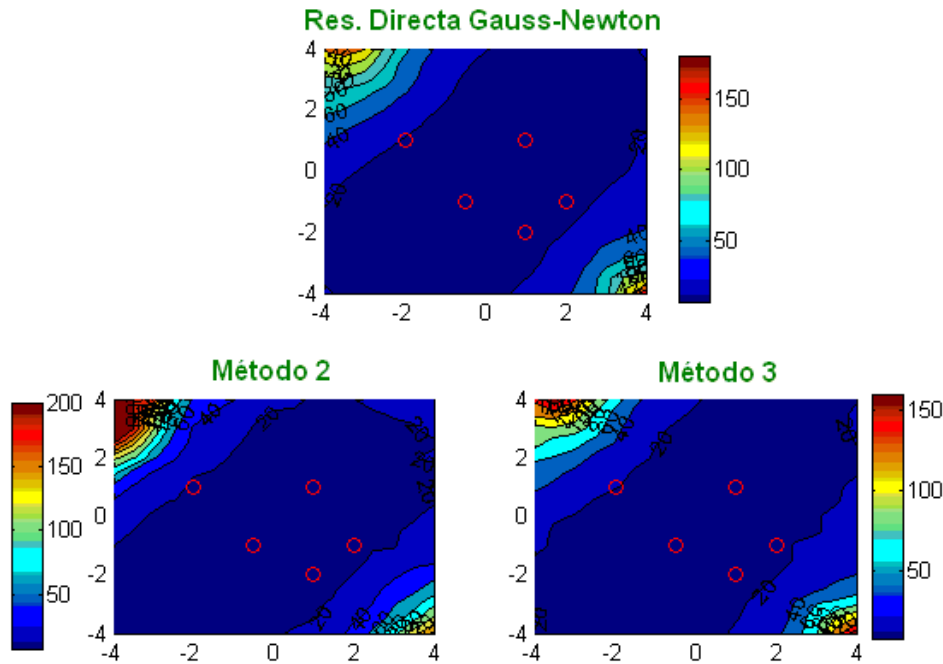


Figura 3.23: Comparativa PDOP obtenido entre los métodos propuestos y la resolución directa por Gauss-Newton, para la cuarta distribución de balizas. Los círculos son las proyecciones de las balizas en el suelo.

En la tabla 3.7 se vuelven a mostrar las diferencias medias y máximas para esta prueba de los diferentes algoritmos. En el caso del segundo método esta vez esas diferencias son menores que en el caso anterior, de unas diferencias medias y máximas de 35,43 % y 513 % se pasa a unas diferencias de 13,20 % y 146 % respectivamente. En el caso del método LTI se siguen manteniendo los porcentajes similares al resto de las pruebas.

Tabla 3.7: Diferencias medias y máximas entre los algoritmos en el cuarto test.

	Método LTR	Método LTI
Dif. Media	13,20 %	1.21 %
Dif. Máxima	146 %	5,21 %

En la figura 3.24 se muestra el histograma con el porcentaje en diferencia de cada uno de los métodos comparado con el de resolución directa mediante Gauss-Newton, las diferencias en el caso del segundo algoritmo son inferiores al 30 % en el 93 % de los puntos, mientras que en el caso del tercer método esas diferencias son inferiores al 2.5 % en el 90 % de los casos. Debido a la diferencia de escala entre ambos métodos en la figura no se han igualado las escalas del eje 'X'.

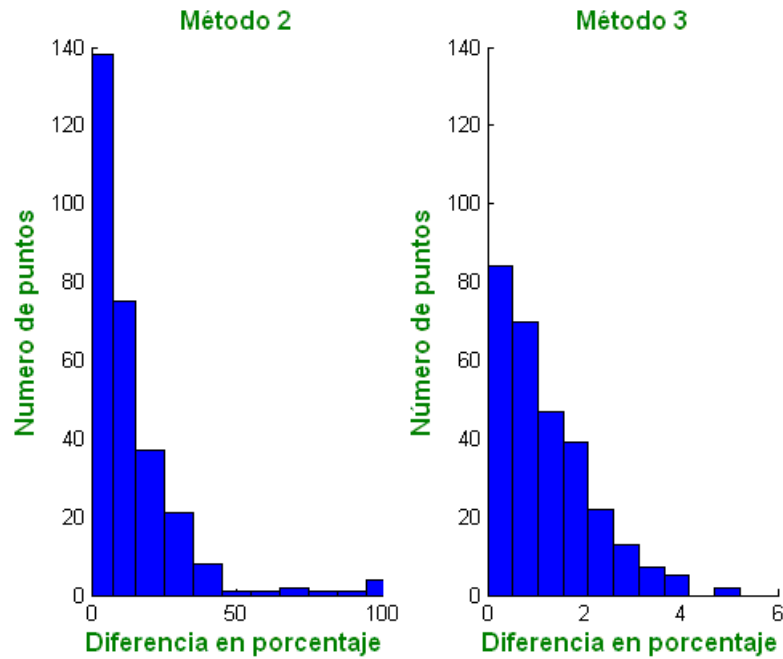


Figura 3.24: Histograma con el porcentaje en diferencia entre los métodos propuestos LTR y LTI y el método de resolución directa para la cuarta distribución de balizas. Izquierda, método LTR Vs. Resolución directa de Gauss-Newton. Derecha, método LTI Vs. Resolución directa Gauss-Newton.

Por último en la tabla 3.8 se muestra el tiempo empleado en cada prueba, que al igual que en los casos anteriores se comprueba que los algoritmos propuestos son mucho más eficientes que el algoritmo de referencia. También se observa que al ser una distribución de balizas peor los tiempos de simulación aumentan respecto a la prueba anterior.

Tabla 3.8: Tiempo necesario para finalizar el cuarto test.

Gauss-Newton	Método LTR	Método LTI
40min. 40sg.	2min. 47sg	4min. 40sg.

3.5. Conclusiones

Este capítulo se ha dedicado al desarrollo de un nuevo algoritmo de posicionamiento hiperbólico basados en el operador de Cayley-Menger con tres métodos de resolución. Este operador ofrece una serie de propiedades matemáticas [TORC03] que resultan muy útiles para el cálculo de la posición de nodos móviles, ya que permiten obtenerla de forma más rápida y con similar precisión que los algoritmos iterativos clásicamente utilizados [YTBJ08].

Los algoritmos propuestos derivan del método para la trilateración esférica propuesto por [TR05]. Este algoritmo presenta la ventaja de que no necesita de metodos de resolución compleja para obtener la posición del nodo móvil ya que al utilizar las propiedades

de este operador consigue encontrar la posición de forma directa. En una primera aproximación se han modificado las ecuaciones para adaptarlas para el caso hiperbólico y posteriormente, analizando las nuevas ecuaciones no lineales, se han propuesto tres nuevas formas de resolución:

- El método LZC (método 1) sólo es empleable si se cumple la restricción geométrica de que todas las balizas se encuentran en el mismo plano, esto es, estén a la misma altura. Bajo esas condiciones se eliminan las no linealidades de las ecuaciones y se puede encontrar la posición del nodo móvil resolviendo un sistema de ecuaciones lineal. Este método en las pruebas realizadas demuestra ser mucho más rápido que el método de resolución directa de las ecuaciones de distancia de Gauss-Newton y con similar precisión.
- El método LTR (método 2) es válido para cualquier distribución de las balizas y a través de una rotación del sistema de referencia, consigue que al menos tres balizas sí estén en el mismo plano convirtiendo de esa forma el sistema de ecuaciones no lineal en un sistema cuadrático. En las pruebas realizadas se ha comprobado la eficiencia computacional del método pero también se ha constatado que para determinadas configuraciones puede ofrecer una precisión mucho peor que los demás algoritmos.
- Por último el método LTI (método 3) obtiene la posición resolviendo el sistema de ecuaciones por minimización de una función de coste no lineal. En los diferentes tests se ha verificado que este algoritmo es algo mas lento que los dos anteriores pero más rápido que el método de resolución directa mediante Gauss-Newton y que ofrece una precisión similar a este último.

Algoritmo para autocalibración de un sistema LPS balizado

En este capítulo se abordan los algoritmos propuestos para realizar una calibración de balizas dentro de un sistema LPS. Primero se analiza el método propuesto en el caso de emplear un posicionamiento esférico (trilateración esférica), para a continuación incidir sobre la versión del algoritmo en el caso de que se utilice un posicionamiento hiperbólico (trilateración hiperbólica) [DBS96]. Este segundo método de posicionamiento presenta la ventaja de que para realizar las medidas no se necesita ningún mecanismo de sincronización entre las balizas y el nodo móvil a localizar, aunque como la calibración es un proceso realizado ocasionalmente, esto no es tan problemático como en la fase de navegación

4.1. Introducción

Como se ha mencionado en el capítulo 2 para poder realizar la calibración de las balizas de un sistema LPS, es necesario tomar medidas de distancia a las mismas desde distintos puntos de test dentro del área de cobertura. Si se conociera la posición de estos puntos sería posible encontrar la posición de las balizas simplemente empleando el algoritmo de posicionamiento de modo inverso [MF99]. Estos métodos tienen el inconveniente de tener que medir de forma manual la posición de todos los puntos de prueba.

En los algoritmos que se proponen a continuación no es necesario conocer la ubicación de todos estos puntos de test, tan sólo se debe conocer la ubicación de tres de ellos. El primero define el origen del sistema de referencia, esto elimina ambigüedades en la solución debidas a las traslaciones del sistema de coordenadas. Los otros dos puntos eliminan ambigüedades en la solución debidas a la rotación del sistema de referencia.

Para los algoritmos propuestos se necesitan como datos iniciales:

- Un vector que contiene los valores asignados inicialmente a cada una de las variables del algoritmo, **pai**, en este caso las coordenadas iniciales de las N balizas, $(\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi})$ con $i = 1 \dots N$ y la localización de los M puntos de test desconocidos

(\hat{x}_j, \hat{y}_j) con $j = 1 \dots M$. Como se puede ver, no se incluye la componente z de los puntos de test desconocidos ya que se supone conocida para el algoritmo.

$$\mathbf{pai} = (\hat{b}_{x1}, \hat{b}_{y1}, \hat{b}_{z1}, \dots, \hat{b}_{zN}, \hat{x}_1, \hat{y}_1, \dots, \hat{x}_M, \hat{y}_M)$$

- La distancia entre los puntos de test y las balizas. En el caso de sistemas basados en trilateración esférica es la distancia absoluta entre ambos elementos (r_i). En el caso de sistemas con trilateración hiperbólica es la distancia existente desde cada baliza y el nodo en cuestión menos la distancia desde el nodo a la baliza tomada como referencia (Δr_i).
- La localización de los tres puntos de test conocidos, que evitan las ambigüedades en las soluciones $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$

En la figura. 4.1 se muestra un esquema general del proceso de autocalibración.

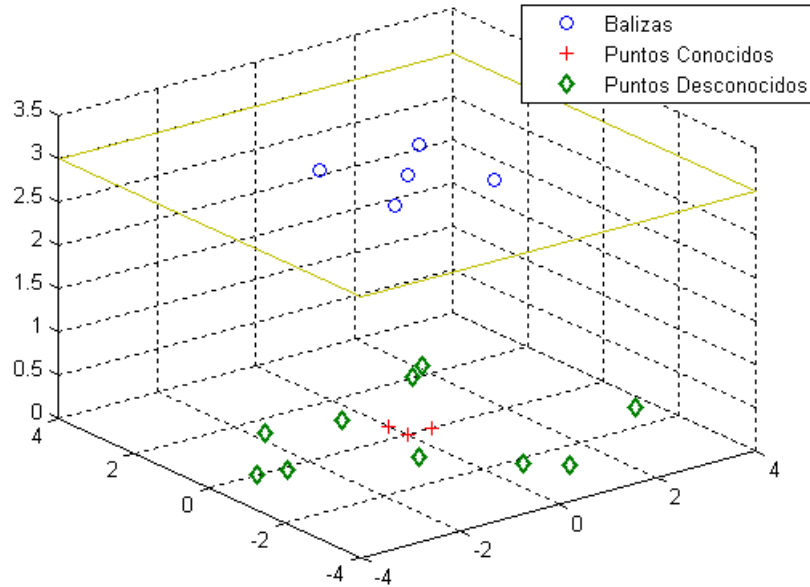


Figura 4.1: Esquema de autocalibración. Ejemplo de ubicación de los puntos de test desconocidos y conocidos para realizar la calibración del sistema LPS.

Este capítulo se ha estructurado de la siguiente manera: en la sección 4.2 se muestra el método propuesto para realizar la autocalibración tanto si se emplea un algoritmo de trilateración esférica como hiperbólica; a continuación en el apartado 4.3 se detalla un algoritmo que permite mejorar los resultados obtenidos con los algoritmos de autocalibración; en la sección 4.4 se validan y se comprueban los resultados obtenidos mediante distintas pruebas realizadas bajo simulación; por último en la sección 4.5 se detallan las conclusiones extraídas en este capítulo.

4.2. Algoritmo propuesto para la autocalibración de balizas

4.2.1. Caso esférico

En esta sección se expone el algoritmo propuesto para realizar una calibración en sistemas que utilicen trilateración esférica.

Si se define r_{im} como la distancia real entre la baliza i de coordenadas (b_{xi}, b_{yi}, b_{zi}) y el punto de test m de coordenadas (x_m, y_m, z_m) . Su valor viene dado por:

$$r_{im} = \sqrt{(b_{xi} - x_m)^2 + (b_{yi} - y_m)^2 + (b_{zi} - z_m)^2} \quad (4.1)$$

Por otro lado, si se define \hat{r}_{im} como la distancia entre la posición estimada de la baliza i de coordenadas $(\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi})$ y la posición estimada del punto de test m con coordenadas $(\hat{x}_m, \hat{y}_m, \hat{z}_m)$:

$$\hat{r}_{im} = \sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} \quad (4.2)$$

El algoritmo utiliza el método iterativo de Gauss-Newton [Fle87] para minimizar la suma de los errores cuadráticos que existe entre las distancias medidas y las distancias obtenidas con los valores estimados de cada una de las incógnitas del sistema mediante la siguiente expresión:

$$F = \sum_{i=1}^N \sum_{m=4}^M (\hat{r}_{im} - r_{im})^2 = \sum_{i=1}^N \sum_{m=4}^M [f_{im}]^2 \quad (4.3)$$

Donde:

$$f_{im} = \sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} - r_{im} \quad (4.4)$$

En el sumatorio anterior se hace variar el índice m entre 4 y M , ya que los puntos $m = 1, 2, 3$ corresponden a los puntos de test cuya posición es conocida.

Derivando la expresión anterior respecto a todas las incógnitas del sistema, partiendo de que se conoce la altura de todos los puntos de test y reorganizando el sistema en la forma $\mathbf{A} \cdot \Delta \mathbf{X} = \mathbf{B}$ se obtiene:

$$\begin{aligned}
\mathbf{A} &= \begin{pmatrix} \frac{\partial f_{11}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial f_{11}}{\partial \hat{b}_{zN}} & \frac{\partial f_{11}}{\partial \hat{x}_4} & \dots & \frac{\partial f_{11}}{\partial \hat{y}_M} \\ \frac{\partial f_{12}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial f_{12}}{\partial \hat{b}_{zN}} & \frac{\partial f_{12}}{\partial \hat{x}_4} & \dots & \frac{\partial f_{12}}{\partial \hat{y}_M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{NM}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial f_{NM}}{\partial \hat{b}_{zN}} & \frac{\partial f_{NM}}{\partial \hat{x}_4} & \dots & \frac{\partial f_{NM}}{\partial \hat{y}_M} \end{pmatrix} \\
\Delta \mathbf{X} &= [\hat{b}_{x1}, \dots, \hat{b}_{x1}, \hat{x}_4, \dots, \hat{y}_m]^T \\
\mathbf{B} &= [f_{11}, \dots, f_{NM}]^T
\end{aligned} \tag{4.5}$$

Donde:

$$\begin{aligned}
\frac{\partial f_{im}}{\partial \hat{b}_{xi}} &= \frac{\hat{b}_{xi} - \hat{x}_m}{\hat{r}_{im}}, & \frac{\partial f_{im}}{\partial x_m} &= \frac{\hat{x}_m - \hat{b}_{xi}}{\hat{r}_{im}} \\
\frac{\partial f_{im}}{\partial \hat{b}_{yi}} &= \frac{\hat{b}_{yi} - \hat{y}_m}{\hat{r}_{im}}, & \frac{\partial f_{im}}{\partial y_m} &= \frac{\hat{y}_m - \hat{b}_{yi}}{\hat{r}_{im}} \\
\frac{\partial f_{im}}{\partial \hat{b}_{zi}} &= \frac{\hat{b}_{zi} - \hat{z}_m}{\hat{r}_{im}}, & \frac{\partial f_{im}}{\partial z_m} &= \frac{\hat{z}_m - \hat{b}_{zi}}{\hat{r}_{im}}
\end{aligned} \quad \text{con} \begin{cases} i = 1 \dots N \\ m = 4 \dots M \end{cases} \tag{4.6}$$

Resolviendo el sistema por mínimos cuadrados se obtiene:

$$\Delta \mathbf{X} = (\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{B} \tag{4.7}$$

Una vez obtenidos los incrementos de posición se actualiza el vector de posición estimada de la forma:

$$\mathbf{pai}_{k+1} = \mathbf{pai}_k - \Delta \mathbf{X}_k$$

Este proceso se repite hasta que los $\Delta \mathbf{X}$ sean menores que un umbral previamente fijado. Para las pruebas que se realizan en esta tesis, el umbral se ha fijado en $\Delta \mathbf{X}_k \leq 10^{-6}$, además si a las 100 iteraciones el algoritmo no ha convergido se cambian las condiciones iniciales y se reinicia el algoritmo.

4.2.2. Caso hiperbólico

En este caso, en vez de la distancia absoluta entre los puntos de test y las balizas, se mide el tiempo entre la recepción de la señal de una baliza de referencia y las señales del resto de las balizas. Multiplicando este tiempo por la velocidad de los ultrasonidos en el aire, se obtiene el incremento de distancia deseado (Δr_i). Para el desarrollo de las

expresiones se va a suponer, sin pérdida de generalidad, que la baliza de referencia es la primera baliza.

Definiendo el incremento de distancia entre la baliza i y el punto m respecto de la primera baliza en el punto m de la forma:

$$\begin{aligned}\Delta r_{im} = r_{im} - r_{1m} = & \sqrt{(b_{xi} - x_m)^2 + (b_{yi} - y_m)^2 + (b_{zi} - z_m)^2} \\ & - \sqrt{(b_{x1} - x_m)^2 + (b_{y1} - y_m)^2 + (b_{z1} - z_m)^2}\end{aligned}\quad (4.8)$$

También se puede definir el incremento de distancia estimada a partir de los puntos estimados como:

$$\begin{aligned}\Delta \hat{r}_{im} = \hat{r}_{im} - \hat{r}_{1m} = & \sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} \\ & - \sqrt{(\hat{b}_{x1} - \hat{x}_m)^2 + (\hat{b}_{y1} - \hat{y}_m)^2 + (\hat{b}_{z1} - \hat{z}_m)^2}\end{aligned}\quad (4.9)$$

Empleando la misma estrategia que en el caso anterior, se trata de minimizar iterativamente el error cuadrático entre los incrementos de distancia medidos y los estimados con la siguiente expresión:

$$G = \sum_{i=2}^N \sum_{m=4}^M (\Delta \hat{r}_{im} - \Delta r_{im})^2 = \sum_{i=2}^N \sum_{m=4}^M [g_{im}]^2 \quad (4.10)$$

Siendo:

$$\begin{aligned}g_{im} = & \left(\sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} \right. \\ & \left. - \sqrt{(\hat{b}_{x1} - \hat{x}_m)^2 + (\hat{b}_{y1} - \hat{y}_m)^2 + (\hat{b}_{z1} - \hat{z}_m)^2} \right) - \Delta r_{im}\end{aligned}\quad (4.11)$$

En este caso se ha hecho que el índice i del sumatorio varíe entre 2 y N, ya que la primera baliza se asume que es la de referencia, y al igual que en el caso esférico el índice m varia entre 4 y M.

Derivando y agrupando los términos como en la autocalibración esférica se llega a:

$$\mathbf{A} = \begin{pmatrix} \frac{\partial g_{21}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial g_{21}}{\partial \hat{b}_{zN}} & \frac{\partial g_{21}}{\partial \hat{x}_4} & \dots & \frac{\partial g_{21}}{\partial \hat{y}_M} \\ \frac{\partial g_{22}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial g_{22}}{\partial \hat{b}_{zN}} & \frac{\partial g_{22}}{\partial \hat{x}_4} & \dots & \frac{\partial g_{22}}{\partial \hat{y}_M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_{NM}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial g_{NM}}{\partial \hat{b}_{zN}} & \frac{\partial g_{NM}}{\partial \hat{x}_4} & \dots & \frac{\partial g_{NM}}{\partial \hat{y}_M} \end{pmatrix}$$

$$\Delta \mathbf{X} = [\hat{b}_{x1}, \dots, \hat{b}_{x1}, \hat{x}_4, \dots, \hat{y}_m]^T$$

$$\mathbf{B} = [f_{11}, \dots, f_{NM}]^T$$
(4.12)

Donde:

$$\begin{aligned} \frac{\partial g_{im}}{\partial b_{xi}} &= \frac{\hat{b}_{xi} - \hat{a}_m}{\hat{r}_{im}}; & \frac{\partial g_{im}}{\partial b_{x1}} &= \frac{\hat{x}_m - \hat{b}_{x1}}{\hat{r}_{1m}}; \\ \frac{\partial g_{im}}{\partial b_{yi}} &= \frac{\hat{b}_{yi} - \hat{a}_m}{\hat{r}_{im}}; & \frac{\partial g_{im}}{\partial b_{y1}} &= \frac{\hat{y}_m - \hat{b}_{y1}}{\hat{r}_{1m}}; \\ \frac{\partial g_{im}}{\partial b_{zi}} &= \frac{\hat{b}_{zi} - \hat{a}_m}{\hat{r}_{im}}; & \frac{\partial g_{im}}{\partial b_{z1}} &= \frac{\hat{z}_m - \hat{b}_{z1}}{\hat{r}_{1m}}; \\ \frac{\partial g_{im}}{\partial x_m} &= \frac{\hat{x}_m - \hat{b}_{xi}}{\hat{r}_{im}} - \frac{\hat{x}_m - \hat{b}_{xi}}{\hat{r}_{im}} & \text{con } \begin{cases} i = 2 \dots N \\ m = 4 \dots M \end{cases} \\ \frac{\partial g_{im}}{\partial y_m} &= \frac{\hat{y}_m - \hat{b}_{yi}}{\hat{r}_{im}} - \frac{\hat{y}_m - \hat{b}_{yi}}{\hat{r}_{im}} \\ \frac{\partial g_{im}}{\partial z_m} &= \frac{\hat{z}_m - \hat{b}_{zi}}{\hat{r}_{im}} - \frac{\hat{z}_m - \hat{b}_{zi}}{\hat{r}_{im}} \end{aligned}$$
(4.13)

Resolviendo el sistema por mínimos cuadrados se obtiene:

$$\Delta \mathbf{X} = (\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{B}$$
(4.14)

El vector que contiene las posiciones estimadas es actualizado de la misma forma que en el caso esférico, el algoritmo se repite hasta que los $\Delta \mathbf{X}$ sean menores que el umbral deseado.

4.2.3. Diagrama de flujo genérico

En este apartado, se muestra el diagrama de flujo genérico para los algoritmos de autocalibración propuestos en las secciones previas, (véase la figura 4.2). Este diagrama

es válido tanto para el algoritmo de trilateración esférica como para el de trilateración hiperbólica, ya que el proceso que siguen ambos métodos es similar y lo único que varía entre ellos es su formulación matemática.

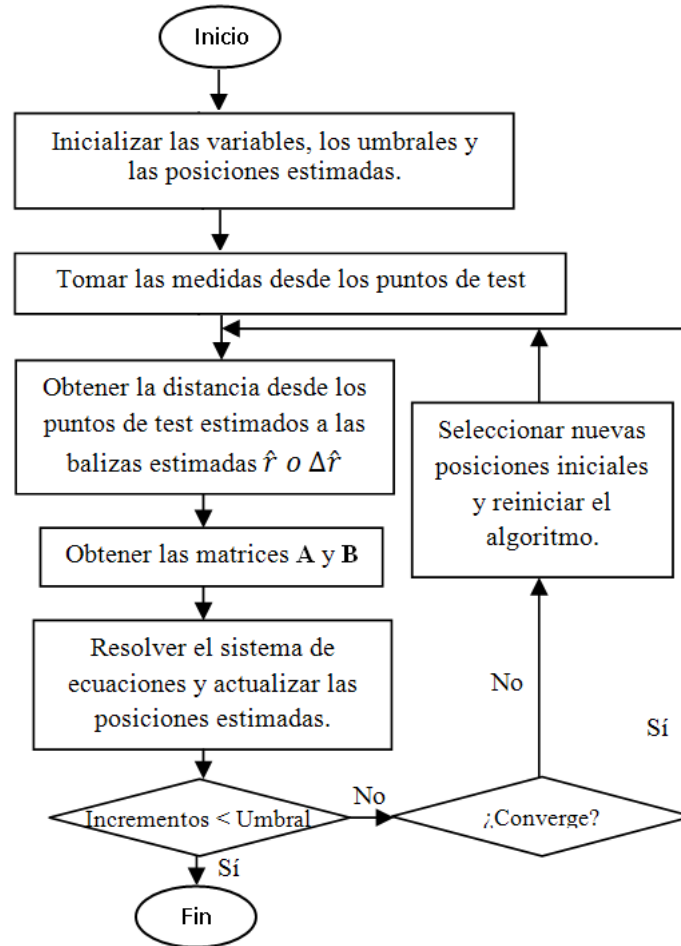


Figura 4.2: Diagrama de flujo de los algoritmos de autocalibración propuestos.

4.3. Puntos óptimos de test

Como se ha descrito anteriormente los puntos de test pueden establecerse en cualquier parte dentro del área de cobertura, pero dependiendo de esta localización, los errores cometidos en la medida pueden afectar en mayor o menor medida a la solución final del sistema. Por ejemplo, ¿es mejor ubicarlos todos juntos debajo de las balizas, en los extremos del área de cobertura o repartidos por todas las zonas equitativamente?

En esta sección se propone un método para encontrar las mejores zonas dentro del área de cobertura para ubicar los puntos de test.

Como se ha explicado en las secciones anteriores, para poder encontrar la localización de las balizas es necesario resolver un sistema del tipo $\mathbf{A} \cdot \Delta\mathbf{X} = \mathbf{B}$ donde $\Delta\mathbf{X}$ es un vector que contiene todas las incógnitas del sistema, para poder minimizar el error en la

localización de las balizas es necesario estudiar la covarianza de $\Delta \mathbf{X}$. Se asume que la matriz \mathbf{A} no tiene porqué ser cuadrada.

$$\begin{aligned} cov(\Delta \mathbf{X}) &= E[\Delta \mathbf{X} \cdot \Delta \mathbf{X}^T] = \mathbf{E} \left[\left((\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{B} \right) \cdot \left((\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{B} \right)^T \right] = \\ &= E \left[(\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{B} \cdot \mathbf{B}^T \cdot \mathbf{A} \cdot (\mathbf{A}^T \cdot \mathbf{A})^{-1} \right] \end{aligned} \quad (4.15)$$

Asumiendo que la covarianza de \mathbf{B} es de la forma $cov(\mathbf{B}) = \mathbf{I}_N \sigma^2$, donde σ_N es la desviación típica de ruido en la medida de distancia entre las balizas y los puntos de test y (suponiendo que es la misma para todos), e \mathbf{I}_N es la matriz identidad, se obtiene:

$$\begin{aligned} cov(\Delta \mathbf{X}) &= E \left[(\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{B} \cdot \mathbf{B}^T \cdot (\mathbf{A}^T \cdot \mathbf{A}) \cdot (\mathbf{A}^T \cdot \mathbf{A})^{-1} \right] = (\mathbf{A}^T \mathbf{A})^{-1} \cdot cov(\mathbf{B}) = \\ &= (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{I}_N \sigma^2 \end{aligned} \quad (4.16)$$

Por lo que la función que hay que minimizar para poder reducir el error en la localización de las balizas y encontrar las zonas óptimas de medida es:

$$h = Trace(\mathbf{A}^T \mathbf{A})^{-1} \quad (4.17)$$

Para minimizar esta función, se utilizan los métodos de minimización de funciones no lineales con restricciones propuestos en [CL93, Pow78]. Estos métodos han sido ampliamente estudiados y existen funciones ya diseñadas para emplearlos en aplicaciones matemáticas como MATLAB. En el Anexo C se detalla el funcionamiento de esta función

Para los métodos propuestos las restricciones en este caso corresponderían con el área de cobertura del sistema LPS. Además es necesario incluir en la iniciación del algoritmo de minimización el número de puntos de medida desconocidos. Las posiciones devueltas por el algoritmo corresponden a las zonas donde se deben ubicar los puntos de test para conseguir la mejor calibración del sistema, esto es, una mejor ubicación de las balizas.

4.4. Resultados de simulación

En esta sección se presentan diferentes pruebas en simulación que muestran el funcionamiento de los algoritmos propuestos en este capítulo. Primero se realizan las pruebas para un posicionamiento esférico y a continuación para un posicionamiento hiperbólico. Las diferentes pruebas se han estructurado de la siguiente manera: se proponen dos con-

figuraciones de balizas diferentes y se obtiene las zonas de puntos óptimos para dichas configuraciones; a continuación se comparan los resultados de autocalibración obtenidos empleando las zonas óptimas o situando los puntos de test desconocidos de forma aleatoria; por último se comprueba como afecta en las autocalibraciones el aumento del número de puntos de test desconocidos. Para la realización de estas pruebas, al igual que en el capítulo anterior, se emplea el PDOP. En el Apéndice B se encuentra más información sobre este parámetro.

4.4.1. Caso esférico

Se realiza la simulación de dos configuraciones diferentes de balizas. En la primera distribución las balizas forman un cinco de dominó ya que es la distribución óptima para cinco balizas en el techo [ML09], esto es, la que menor PDOP tiene cuando se localiza un objeto en el suelo. En cambio, en la segunda distribución las balizas se han situado de forma irregular por la zona de cobertura obteniéndose de esa forma un PDOP peor dentro del área de cobertura. Además se ha definido una zona de cobertura de $8 \times 8 \text{ m}^2$. En la tabla 4.1 se muestran las coordenadas de las balizas para las dos distribuciones.

Tabla 4.1: Distribución de las balizas para las pruebas de autocalibración bajo simulación

	Distribución 1 $[x, y, z]$	Distribución 2 $[x, y, z]$
Baliza 1	$[0.0, 0.0, 3.0] \text{ m.}$	$[-2.7, -2.8, 3.0] \text{ m.}$
Baliza 2	$[2.0, 2.0, 3.0] \text{ m.}$	$[-2.4, -0.7, 3.0] \text{ m.}$
Baliza 3	$[2.0, -2.0, 3.0] \text{ m.}$	$[-0.4, -0.4, 3.0] \text{ m.}$
Baliza 4	$[-2.0, 2.0, 3.0] \text{ m.}$	$[-2.5, -0.3, 3.0] \text{ m.}$
Baliza 5	$[-2.0, -2.0, 3.0] \text{ m.}$	$[-2.9, -1.3, 3.0] \text{ m.}$

En la figura 4.3 se puede observar como en el primer caso el PDOP varía entre 2 y 4 mientras que con la segunda distribución la variación es entre 3 y 12, confirmandose de ese modo que la segunda distribución es peor que la primera.

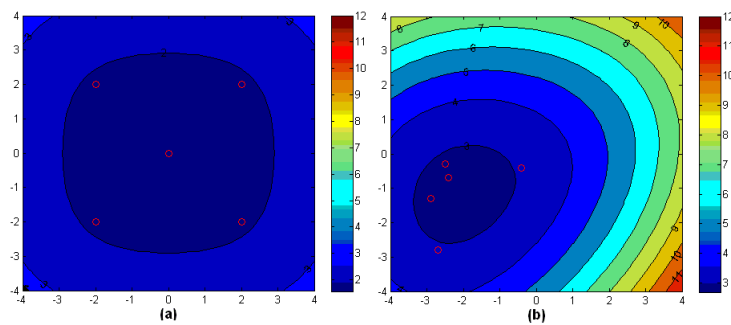


Figura 4.3: PDOP para ambas distribuciones de balizas en el caso esférico. (a) PDOP para $z=0$ en el área de cobertura cuando se simula la primera distribución de balizas. (b) PDOP para $z=0$ en el área de cobertura cuando se simula la segunda distribución de balizas. Los círculos rojos representan la proyección de las balizas en el plano $z=0$.

4.4.1.1. Ubicación de los puntos óptimos

En esta sección se muestra la ubicación de los puntos óptimos si se emplean las distribuciones de balizas anteriormente citadas, se realizarán simulaciones para 4 y 8 puntos de test desconocidos y se ha supuesto que la altura de dichos puntos es $z=0$.

En la tabla 4.2 se muestra la posición de los puntos de test conocidos.

Tabla 4.2: Ubicación de los puntos conocidos para todas las pruebas de autocalibración.

Punto 1 $[x_1, y_1, z_1]$	Punto 2 $[x_2, y_2, z_2]$	Punto 3 $[x_3, y_3, z_3]$
$[0.0, 0.0, 0.0] m.$	$[0.5, 0.0, 0.0] m.$	$[0.0, 0.5, 0.0] m.$

Para facilitar la visualización, en vez de hacer una representación en tres dimensiones, solo se mostrará una representación 2D en el plano de medida ($z=0$). Las proyecciones de las balizas en dicho plano se representan con un círculo azul (○), los puntos conocidos con una cruz roja (+) y los puntos óptimos con un rombo verde (◇).

Después de realizar el algoritmo, en la tabla 4.3 y en la figura 4.4 se puede observar la posición de los puntos óptimos para la primera distribución de balizas. En base a los resultados se puede ver que los puntos óptimos tratan de situarse rodeando las balizas, esto es más claro en la simulación de los 8 puntos óptimos donde forman un círculo alrededor de las balizas.

Tabla 4.3: Coordenadas de los puntos óptimos para la primera distribución de balizas en el caso esférico.

Posición de los puntos óptimos para la primera distribución de balizas		
4 Puntos $[x, y, z] m.$	8 Puntos $[x, y, z] m.$	
$[-4.0, 2.0, 0.0]$	$[2.0, -3.3, 0.0]$	$[4.0, -1.4, 0.0]$
$[-2.8, -1.9, 0.0]$	$[1.6, 4.0, 0.0]$	$[-3.2, -1.9, 0.0]$
$[2.9, 2.8, 0.0]$	$[3.7, 2.2, 0.0]$	$[-3.5, 1.5, 0.0]$
$[-0.0, -4.0, 0.0]$	$[-2.0, 3.7, 0.0]$	$[-1.4, -3.4, 0.0]$

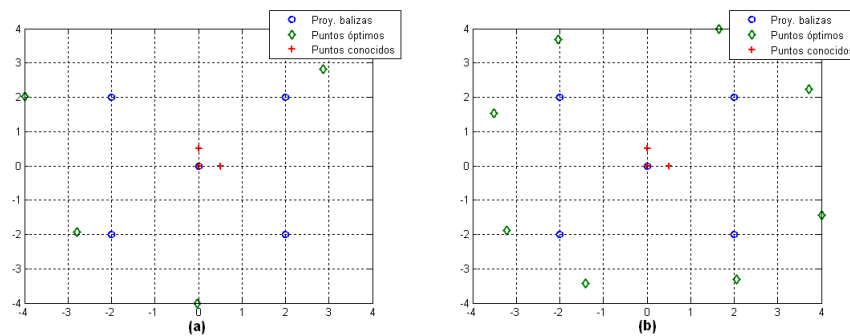


Figura 4.4: Representación de la ubicación de las zonas de puntos óptimos para la primera distribución de balizas en el plano $z=0$ para el caso esférico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.

En las pruebas anteriores se han buscado las zonas óptimas en el plano $z=0$, en el siguiente test se quiere comprobar cómo varía la ubicación de dichos puntos según se incrementa la altura z del plano de medida, ya que se buscará la ubicación de los puntos óptimos desde el plano $z=0$ hasta el plano $z=2$ con pasos de 0.2 m . Los resultados de esta prueba se muestran en la figura 4.5. Se puede comprobar que, según el plano z va subiendo, los puntos óptimos tienden a mantener su estructura pero finalmente tienden a acercarse a las balizas.

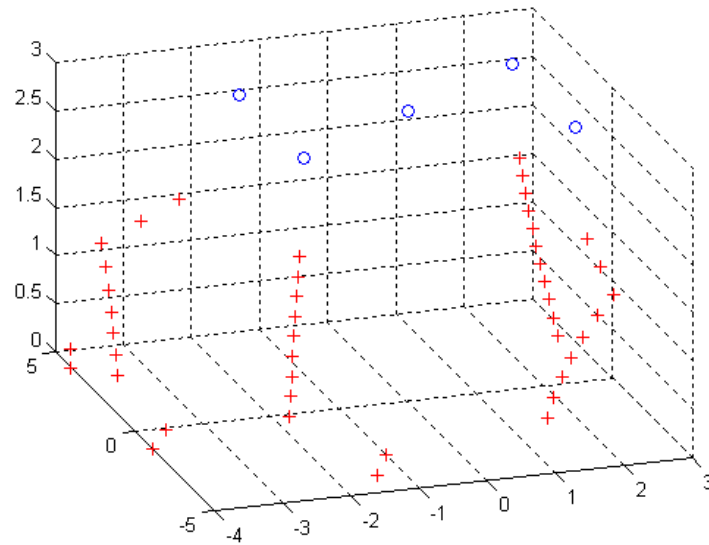


Figura 4.5: Variación de la ubicación de los puntos óptimos con la altura en la primera distribución de balizas para el caso esférico.

Se calculan los puntos óptimos para la segunda distribución de balizas con los mismos parámetros de simulación que en el caso anterior, y se muestra el resultado de la ubicación de los mismos en la tabla 4.4 y en la figura 4.6. Se puede observar que aunque la distribución de los puntos no es tan clara como en el caso anterior, siempre se cumple que algunos puntos de test quedan debajo de la zona de las balizas, y otros en posiciones más alejadas dentro de la zona de cobertura.

Tabla 4.4: Coordenadas de los puntos óptimos para la primera distribución de balizas en el caso esférico.

Posición de los puntos óptimos para la segunda distribución de balizas		
4 Puntos $[x, y, z]\text{ m.}$	8 Puntos $[x, y, z]\text{ m.}$	
$[-4.0, -4.0, 0.0]$	$[-3.6, -3.9, 0.0]$	$[4.0, 1.5, 0.0]$
$[-4.0, -2.2, 0.0]$	$[-0.1, -1.0, 0.0]$	$[-0.4, 4.0, 0.0]$
$[4.0, -4.0, 0.0]$	$[-4.0, -4.0, 0.0]$	$[0.3, -0.9, 0.0]$
$[-1.5, -1.3, 0.0]$	$[-3.6, -1.6, 0.0]$	$[2.4, -0.1, 0.0]$

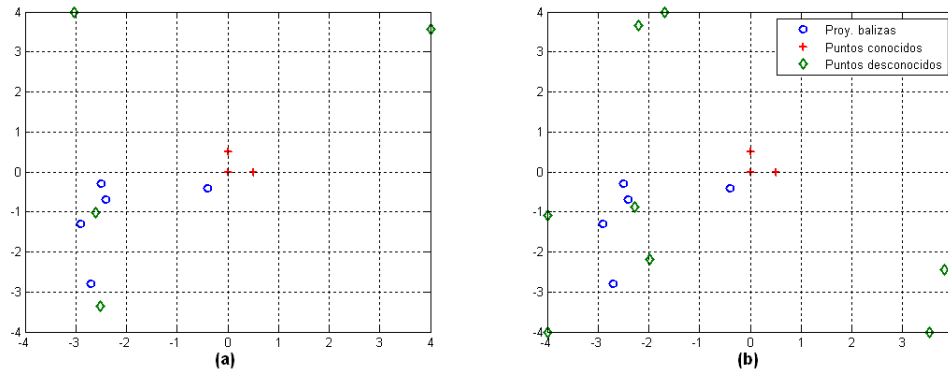


Figura 4.6: Representación de la ubicación de las zonas de puntos óptimos para la segunda distribución de balizas en el plano $z=0$ para el caso esférico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.

En la figura 4.7 se muestran los resultados para el test de variación de la ubicación de los puntos óptimos con la altura. En este caso se observa también que los puntos mantienen su estructura y que también se acercan a la zona de las balizas, según se aumenta la altura del plano donde se ubican los puntos de medida.

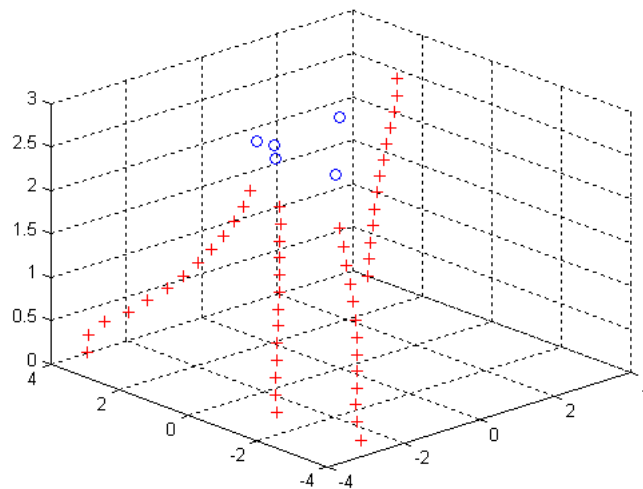


Figura 4.7: Variación de la ubicación de los puntos óptimos con la altura en la segunda distribución de balizas para el caso esférico.

4.4.1.2. Puntos óptimos Vs. Puntos aleatorios

Una vez obtenido los puntos óptimos se comparan los resultados que se obtienen al realizar el proceso de autocalibración en las dos distribuciones de balizas planteadas, ubicando los puntos de test en las zonas óptimas o colocándolos de forma aleatoria por el área de cobertura. Para ello se realiza en cada prueba cien veces la autocalibración del sistema LPS con diferentes valores de ruido y se analiza el error en la ubicación de las balizas. Se simulan tres condiciones diferentes de ruido en las medidas $\sigma = \{0.1, 1, 4\}$ cm.

Para la primera distribución la posición de los puntos de medida desconocidos situados de forma tanto aleatoria como en las zonas óptimas queda reflejada en la tabla 4.5, (en el caso de los puntos óptimos son los calculados en el apartado anterior)

Tabla 4.5: Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la primera distribución de balizas y empleando un posicionamiento esférico.

Posición de los puntos de test desconocidos en la primera prueba	
4 Puntos aleatorios $[x, y, z]$ m.	4 Puntos óptimos $[x, y, z]$ m.
$[-0.7, 3.2, 0.0]$	$[-4.0, 2.0, 0.0]$
$[-3.5, -1.2, 0.0]$	$[-2.8, -1.9, 0.0]$
$[2.5, -3.9, 0.0]$	$[2.9, 2.8, 0.0]$
$[2.9, -2.4, 0.0]$	$[-0.0, -4.0, 0.0]$

En la figura 4.8 se muestran los resultados de la autocalibración para las cien simulaciones con la primera distribución de balizas, se puede observar cómo la dispersión en los resultados aumenta con el nivel de ruido y cómo esta dispersión es menor cuando se emplean puntos óptimos de test.

En el gráfico de la figura 4.9 se puede observar la desviación típica media de las 100 simulaciones en cada una de las componentes $[x, y, z]$ para los diferentes niveles de ruido. Esta desviación típica da idea de la dispersión que existe en los resultados y se obtiene para cada nivel de ruido de la siguiente manera:

- Se ha ejecutado 100 veces el algoritmo de autocalibración para dicho nivel de ruido, obteniéndose de esa forma 100 posiciones diferentes de cada una de las 5 balizas: $[\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi}]_j$ con $i = 1 \dots 100$ y $j = 1 \dots 5$.
- Para cada una de las balizas se calcula la desviación típica de los resultados en cada componente: $[\sigma_x, \sigma_y, \sigma_z]_j$ con $j = 1 \dots 5$.
- Por último se calcula para cada componente la desviación media debido a las 5 balizas: $\sigma_{xmedia} = \frac{1}{5} \sum_{j=1}^5 [\sigma_x]_j$; $\sigma_{ymedia} = \frac{1}{5} \sum_{j=1}^5 [\sigma_y]_j$; $\sigma_{zmedia} = \frac{1}{5} \sum_{j=1}^5 [\sigma_z]_j$

Se puede ver que al aumentar el nivel de ruido en las medidas, aumenta la desviación típica en los resultados y cómo la componente z es la menos afectada de todas. También se muestra cómo los resultados obtenidos con los puntos óptimos son mejores, esto es, la desviación típica en las soluciones es menor que en el caso de emplear puntos aleatorios.

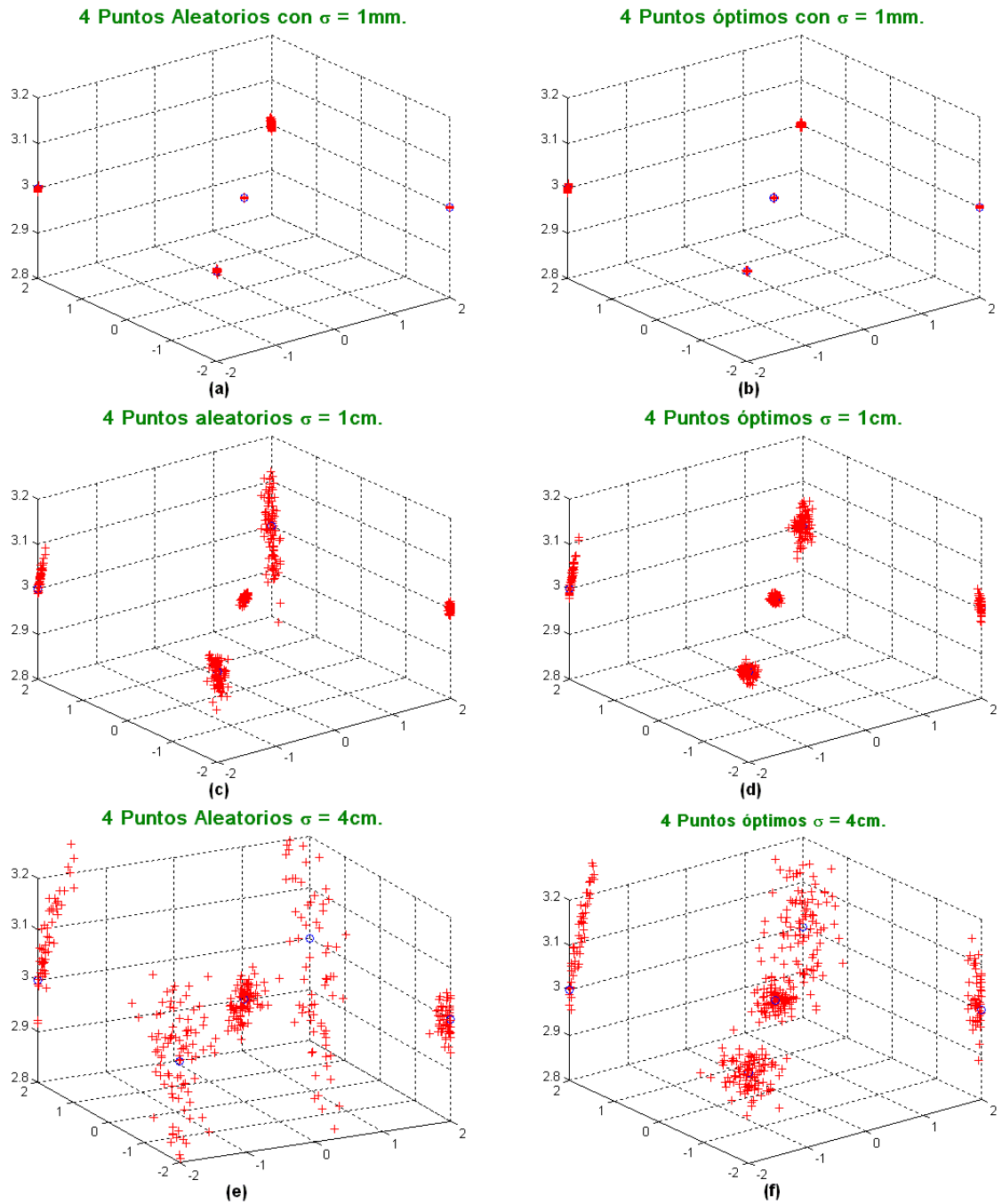


Figura 4.8: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\} \text{ cm}$ situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la primera distribución de balizas con posicionamiento esférico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.

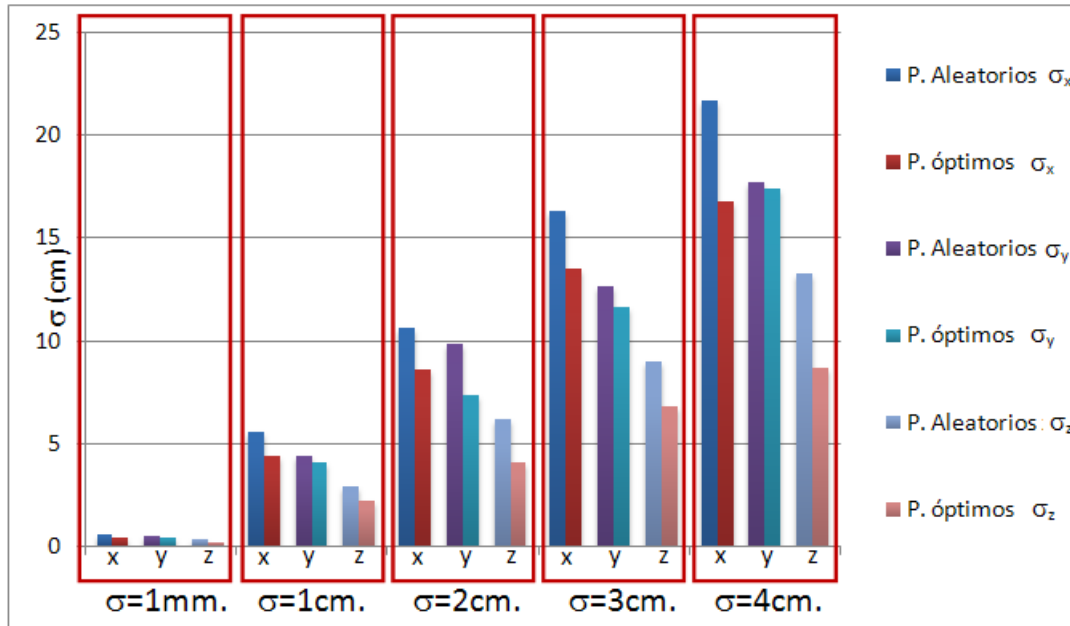


Figura 4.9: Comparación de la desviación típica en cada una de las componentes (x, y, z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la primera distribución de balizas con posicionamiento esférico.

Por último en la tabla 4.6 se muestra el error medio en centímetros de cada componente en la localización de las balizas después de las 100 simulaciones. Este error se calcula de la siguiente forma:

- Se ha realizado 100 veces el algoritmo de autocalibración para dicho nivel de ruido, obteniéndose de esa forma 100 posiciones diferentes de cada una de las balizas: $[\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi}]_j$ con $i = 1 \dots 100$ y $j = 1 \dots 5$.
- Se calcula la posición media de cada baliza con los 100 resultados: $[\hat{b}_{xmedia}, \hat{b}_{ymedia}, \hat{b}_{zmedia}]_j$ con $j = 1 \dots 5$.
- Se calcula el error cometido en la estimación de cada baliza:

$$\mathbf{e}_j = [e_x, e_y, e_z]_j = [\hat{b}_{xmedia}, \hat{b}_{ymedia}, \hat{b}_{zmedia}]_j - [b_x, b_y, b_z]_j$$
- Se calcula el error medio en cada componente con las 5 balizas:

$$e_{xmedio} = \frac{1}{5} \sum_{j=1}^5 [e_x]_j; e_{ymedio} = \frac{1}{5} \sum_{j=1}^5 [e_y]_j; e_{zmedio} = \frac{1}{5} \sum_{j=1}^5 [e_z]_j$$

En este caso se vuelve a observar que los resultados con puntos de test óptimos son mejores que los resultados obtenidos empleando puntos aleatorios, ya que se comete menos error a la hora de ubicar las balizas.

Tabla 4.6: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), empleando puntos aleatorios y óptimos con la primera distribución de balizas y posicionamiento esférico.

	Puntos Aleatorios (cm)			Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.02	0.03	0.01	0.01	0.19	0.01
$\sigma = 1 \text{ cm.}$	0.39	0.33	0.33	0.23	0.18	0.06
$\sigma = 2 \text{ cm.}$	0.53	0.48	0.45	0.41	0.36	0.12
$\sigma = 3 \text{ cm.}$	0.69	0.82	0.79	0.62	0.75	0.31
$\sigma = 4 \text{ cm.}$	1.08	0.94	0.88	0.83	0.80	0.66

De estos resultados se puede concluir que para trilateración esférica los resultados son muy parecidos en el caso de emplear puntos óptimos o puntos de test aleatorios siempre que el ruido en las medidas no sea muy elevado $\sigma \leq 1 \text{ cm.}$ pero para niveles de ruido más elevados es conveniente emplear puntos de test en las zonas óptimas.

Se realizan las mismas pruebas esta vez para la segunda distribución de balizas. En este caso en la tabla 4.7 se muestra la ubicación de los puntos desconocidos tanto si se posicionan de forma aleatoria o en las zonas óptimas.

Tabla 4.7: Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la segunda distribución de balizas y empleando un posicionamiento esférico.

Posición de los puntos de test desconocidos en la segunda prueba	
4 Puntos aleatorios $[x, y, z] \text{ m.}$	4 Puntos óptimos $[x, y, z] \text{ m.}$
[1.0, 3.5, 0.0]	[-4.0, -4.0, 0.0]
[0.9, -1.7, 0.0]	[-4.0, -2.2, 0.0]
[0.1, 3.5, 0.0]	[4.0, -4.0, 0.0]
[3.7, -0.2, 0.0]	[-1.5, -1.3, 0.0]

En la figura 4.10 se muestran los resultados de la autocalibración para las cien simulaciones en la segunda distribución de balizas, se puede observar como al aumentar el nivel de ruido en las medidas, la dispersión en las soluciones también aumenta y como para desviaciones típicas de ruido de 4 cm. en las medidas la dispersión en las soluciones empieza a ser muy elevada en ambos casos.

En la figura 4.11 se puede observar la desviación típica de las 100 simulaciones en cada una de las componentes $[x, y, z]$ para los diferentes niveles de ruido. Se puede comprobar también en este caso que las desviaciones típicas en la solución se ven reducidas ligeramente si se ubican los puntos de medida en las zonas óptimas.

Por último al igual que en el caso anterior se muestran en la tabla 4.8 los errores medios, en centímetros, en cada componente como resultado de las 100 autocalibraciones. En este caso se puede observar como los errores sí que disminuyen considerablemente, sobre todo en las coordenadas (x, z) si se emplean las zonas óptimas, llegando a reducirse casi en un 30 % para desviaciones típicas de ruido en las medidas de 4 cm.

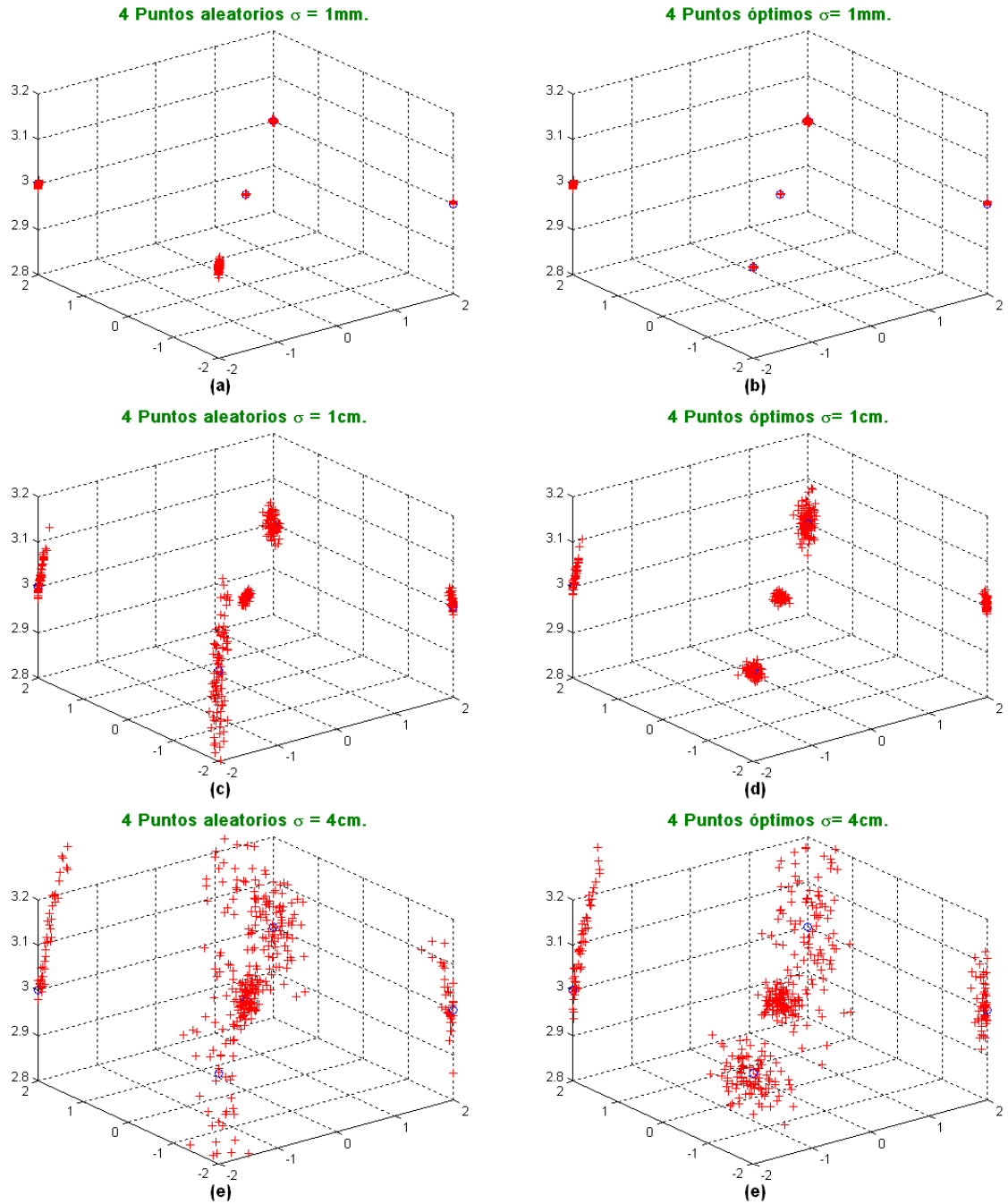


Figura 4.10: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\} \text{ cm}$ situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la segunda distribución de balizas con posicionamiento esférico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.

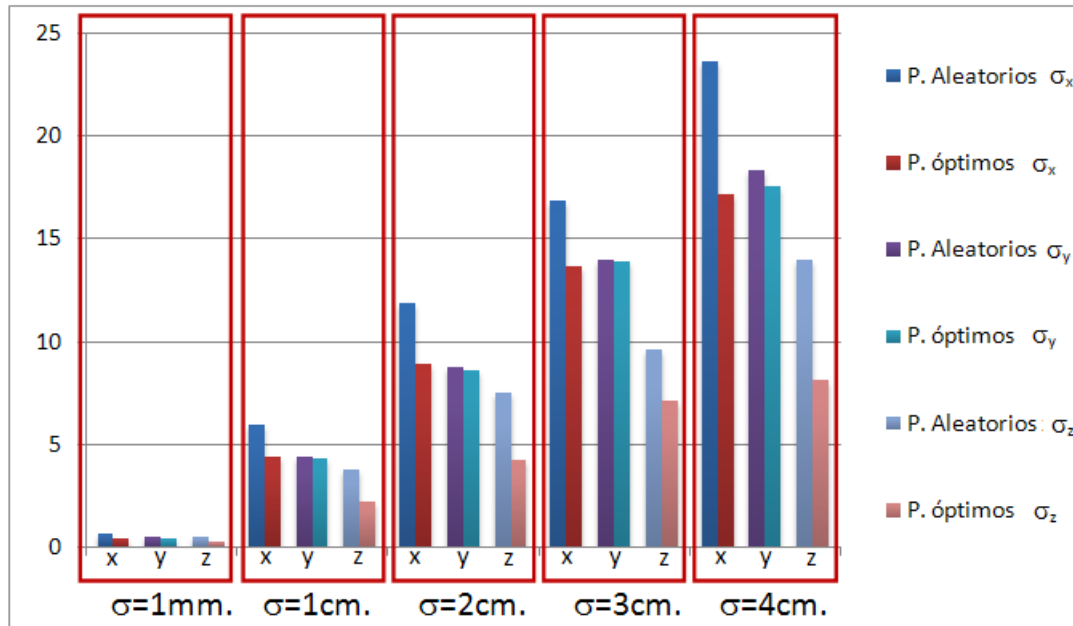


Figura 4.11: Comparación de la desviación típica en cada una de las componentes (x,y,z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la segunda distribución de balizas con posicionamiento esférico.

Tabla 4.8: Error medio en la localización de las balizas por componentes $[x,y,z]$ (cm), empleando puntos aleatorios y óptimos con la segunda distribución de balizas y posicionamiento esférico.

	Puntos Aleatorios (cm)			Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.05	0.03	0.03	0.01	0.02	0.01
$\sigma = 1 \text{ cm.}$	0.62	0.69	0.34	0.44	0.28	0.26
$\sigma = 2 \text{ cm.}$	1.08	1.08	0.91	0.55	0.37	0.28
$\sigma = 3 \text{ cm.}$	2.00	1.42	1.04	0.74	0.86	0.46
$\sigma = 4 \text{ cm.}$	2.90	1.63	2.14	0.94	1.27	0.72

4.4.1.3. Número de puntos de test

En esta prueba se comprueba como varía el error en la localización de las balizas con el aumento de los puntos desconocidos de test para las dos distribuciones de balizas propuestas. En este caso sólo se emplean puntos en zonas óptimas debido a su mejor comportamiento y se emplean los mismos puntos de test que se mostraron en la tablas 4.3 y 4.4. Las condiciones de simulación son las mismas que en el caso anterior.

En la figura 4.12 se observa el resultado de las 100 calibraciones. Para la primera distribución de balizas se comprueba cómo los resultados empleando más puntos óptimos mejoran, ya que la dispersión de las soluciones se reduce considerablemente.

Como se puede observar en la gráfica de la figura 4.13 la dispersión en las 100 auto-calibraciones se reduce considerablemente, sobre todo en el eje Z donde esta reducción

llega a ser de un 75 % de la original. La reducción en las componentes x e y también es apreciable; por ejemplo para una desviación típica de ruido en las medidas de 4 cm. se pasa de 16.81 cm y 17.40 cm a 12.35 cm y 12.51 cm lo que supone una reducción superior al 25 %.

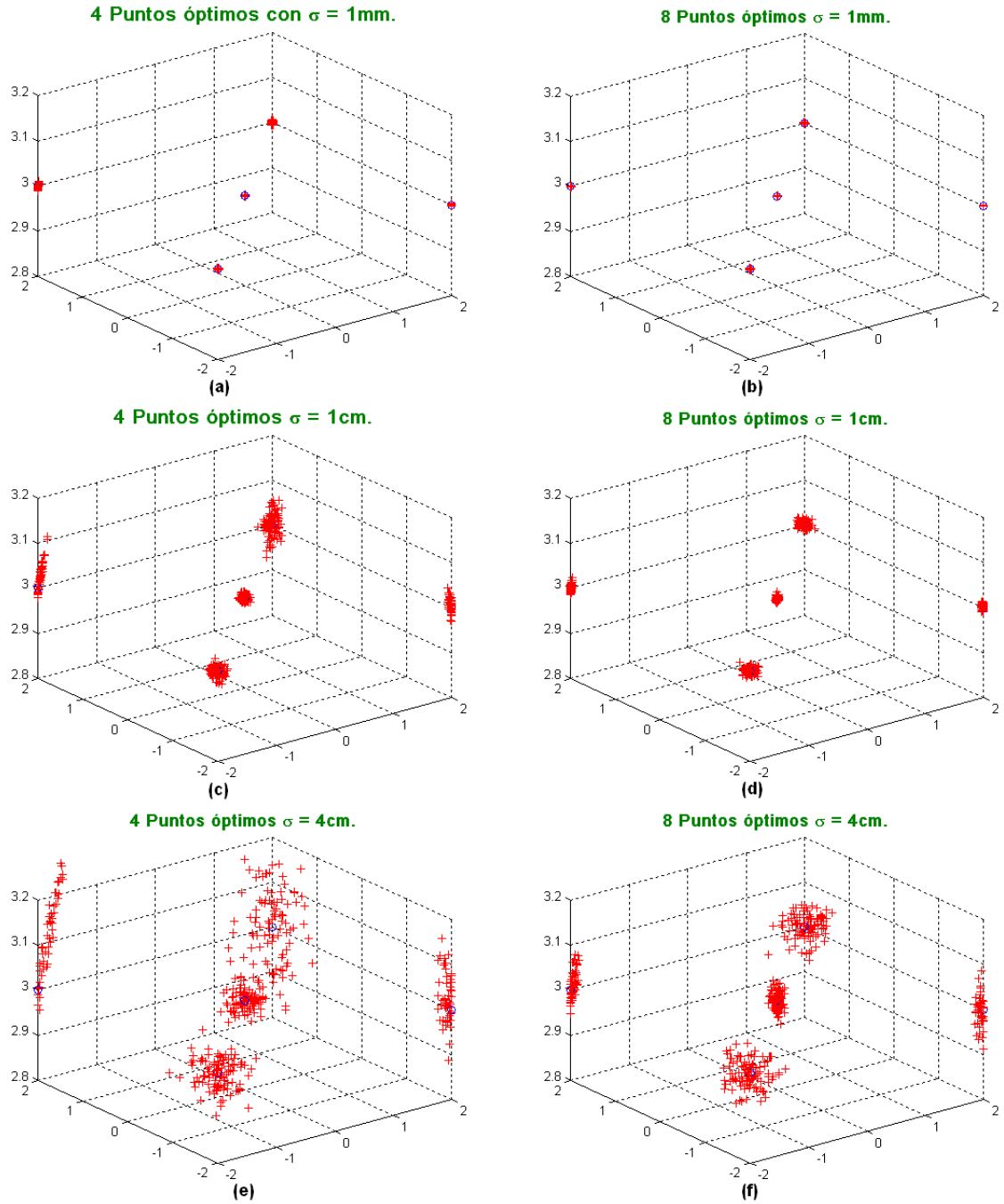


Figura 4.12: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\}$ cm situando los puntos de test desconocidos de forma óptima y empleando la primera distribución de balizas con posicionamiento esférico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima.

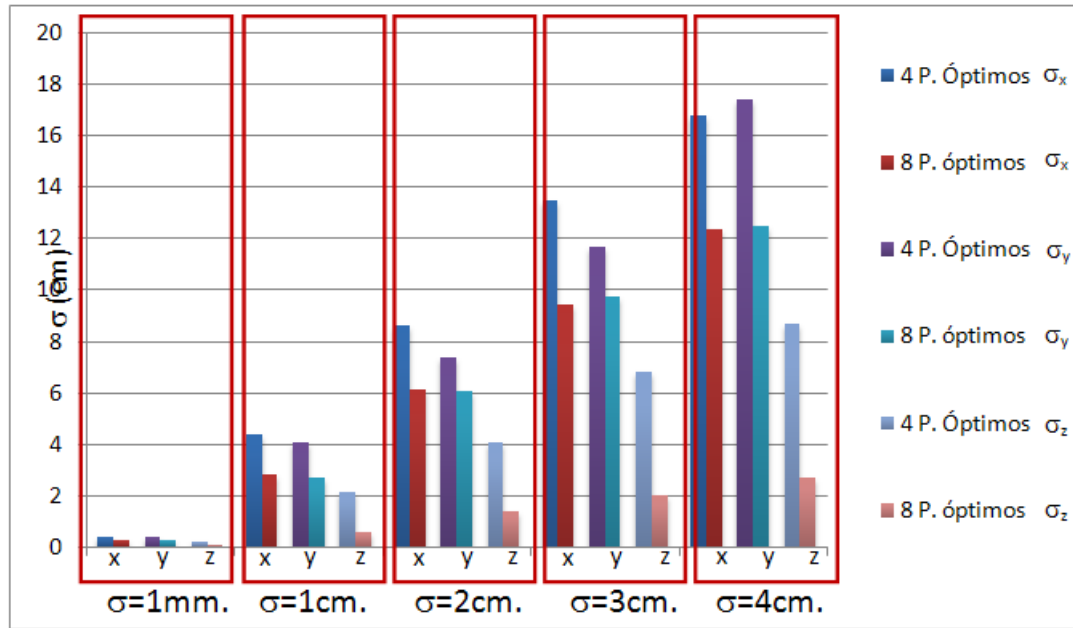


Figura 4.13: Comparación de la desviación típica en cada una de las componentes (x, y, z) , aumentando los puntos de medida, para la primera distribución de balizas con posicionamiento esférico.

Por último en la tabla 4.9 se observa el error medio en cada componente obtenido con las 100 autocalibraciones. Se comprueba que al aumentar el número de puntos de test el error cometido en la ubicación de las balizas se reduce apreciablemente. Por ejemplo para una σ en las medidas de 4 cm el error en la componente z disminuye en más de un 60 % mientras que en las componentes x e y disminuye en torno al 20 %.

Tabla 4.9: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), variando el número de puntos de test con la primera distribución de balizas y posicionamiento esférico.

	4 Puntos Óptimos (cm)			8 Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.01	0.19	0.01	0.01	0.01	0.01
$\sigma = 1 \text{ cm.}$	0.23	0.18	0.06	0.15	0.13	0.06
$\sigma = 2 \text{ cm.}$	0.41	0.36	0.12	0.34	0.20	0.09
$\sigma = 3 \text{ cm.}$	0.62	0.75	0.31	0.42	0.58	0.12
$\sigma = 4 \text{ cm.}$	0.83	0.80	0.66	0.60	0.69	0.19

En la figura 4.14 se muestra para la segunda distribución de balizas el resultado de las 100 calibraciones y se vuelve a comprobar cómo los resultados mejoran empleando más puntos en las zonas óptimas ya que la dispersión de las soluciones se reduce de forma apreciable, sobre todo en los ejes x y z donde esa disminución esta en torno al 50 %, (ver figura 4.15).

En la tabla 4.10 se reflejan los errores medios cometidos en cada una de las componentes, se puede observar cuantitativamente la disminución en el error comentada ante-

riormente, se puede observar que la reducción en los ejes x y z está en torno al 50%, mientras, en el eje y esta reducción es menor.

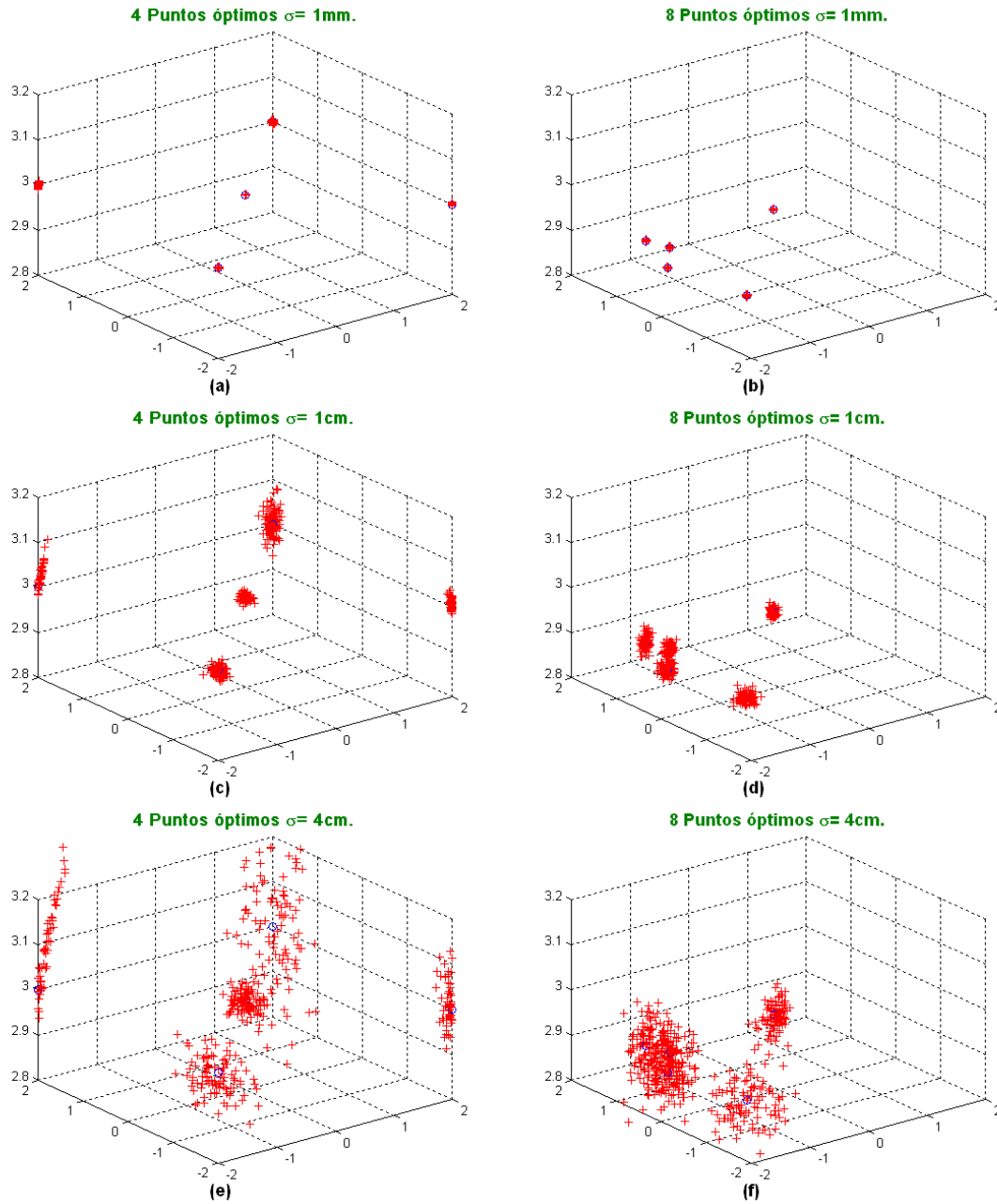


Figura 4.14: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 4\} \text{ cm}$ situando los puntos de test desconocidos de forma óptima y empleando la primera distribución de balizas con posicionamiento esférico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima.

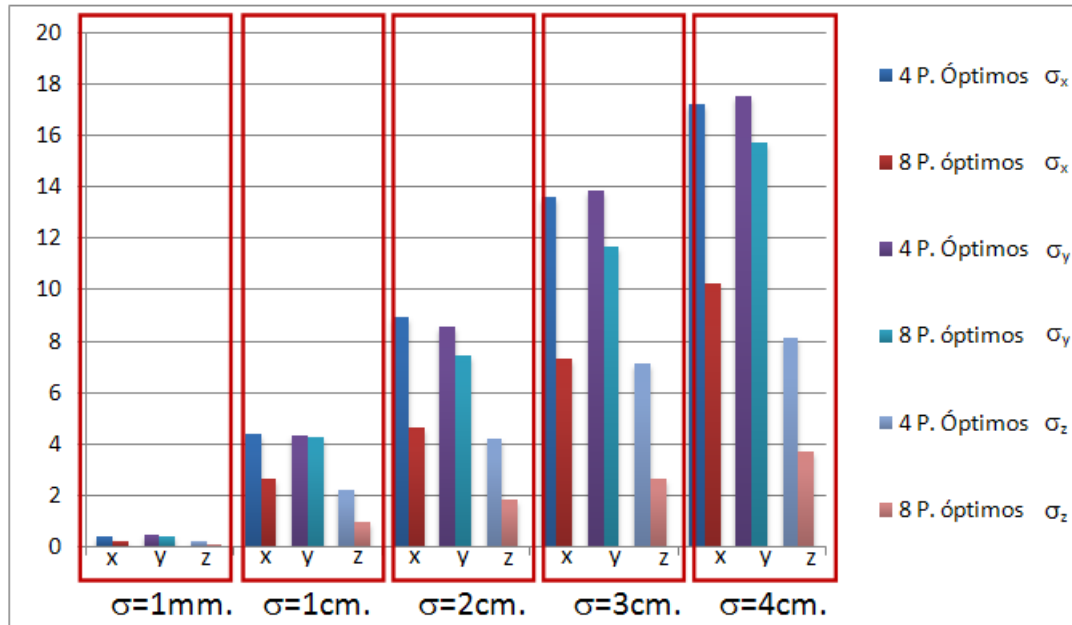


Figura 4.15: Comparación de la desviación típica en cada una de las componentes (x, y, z) , aumentando los puntos de medida, para la segunda distribución de balizas con posicionamiento esférico.

Tabla 4.10: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), variando el número de puntos de test con la segunda distribución de balizas y posicionamiento esférico.

	4 Puntos Óptimos (cm)			8 Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.01	0.02	0.01	0.01	0.01	0.01
$\sigma = 1 \text{ cm.}$	0.44	0.28	0.26	0.14	0.27	0.08
$\sigma = 2 \text{ cm.}$	0.55	0.37	0.28	0.22	0.36	0.14
$\sigma = 3 \text{ cm.}$	0.74	0.86	0.46	0.41	0.74	0.20
$\sigma = 4 \text{ cm.}$	0.94	1.27	0.72	0.64	1.02	0.47

4.4.2. Caso hiperbólico

Una vez comprobado el funcionamiento de los algoritmos en el caso esférico, en esta sección se realizan las mismas pruebas para el caso de la trilateración hiperbólica con las mismas distribuciones de balizas de la tabla 4.1.

En la figura 4.16 se muestra la distribución del PDOP dentro del área de cobertura empleando trilateración hiperbólica. Se puede apreciar que en este caso el PDOP es mucho peor que en el caso esférico, para la primera distribución el PDOP pasa de estar de entre 2 y 4 a oscilar entre 2.5 y 8, y en la segunda distribución el empeoramiento es mucho más acusado ya que el PDOP pasa de oscilar entre 3 y 12 a variar entre 20 y 240. Esto lleva a pensar que los resultados obtenidos con las autocalibraciones serán mucho peores que en el caso esférico.

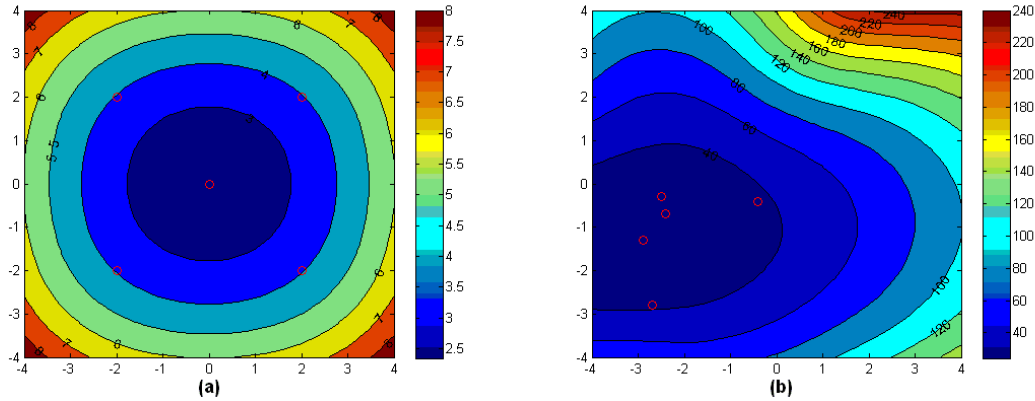


Figura 4.16: PDOP para ambas distribuciones de balizas en el caso hiperbólico. (a) PDOP para $z=0$ en el área de cobertura cuando se simula la primera distribución de balizas. (b) PDOP para $z=0$ en el área de cobertura cuando se simula la segunda distribución de balizas. Los círculos rojos representan la proyección de las balizas en el plano $z=0$.

4.4.2.1. Ubicación de los puntos óptimos

Al igual que en el caso esférico se calculan las zonas de puntos óptimos para ambas distribuciones de balizas. La posición de los puntos de test conocidos seguirán siendo las mismas que en las pruebas anteriores (ver tabla 4.2).

En la tabla 4.11 y en la figura 4.17 se muestra la ubicación de los puntos óptimos para la primera distribución de balizas. Se puede ver claramente cómo al aumentar el número de puntos, éstos tienden a ubicarse rodeando las balizas pero formando un cuadrado, no un círculo como en el caso esférico.

Tabla 4.11: Coordenadas de los puntos óptimos para la primera distribución de balizas en el caso hiperbólico.

Posición de los puntos óptimos para la primera distribución de balizas		
4 Puntos $[x, y, z]$ m.	8 Puntos $[x, y, z]$ m.	
$[-0.25, 4.0, 0.0]$	$[-4.0, -4.0, 0.0]$	$[0.0, -4.0, 0.0]$
$[4.0, -4.0, 0.0]$	$[4.0, 0.0, 0.0]$	$[4.0, 4.0, 0.0]$
$[4.0, -0.69, 0.0]$	$[-4.0, 4.0, 0.0]$	$[4.0, -4.0, 0.0]$
$[-4.0, -4.0, 0.0]$	$[0.0, 4.0, 0.0]$	$[-4.0, 0.0, 0.0]$

Para la segunda distribución de balizas, la ubicación de los puntos óptimos se muestra en la tabla 4.12 y en la figura 4.18. Se puede ver cómo en este caso, al aumentar el número de puntos óptimos, éstos tienden a situarse en el lado donde se concentran las balizas, aunque dos de ellos se sitúan en las esquinas opuestas.

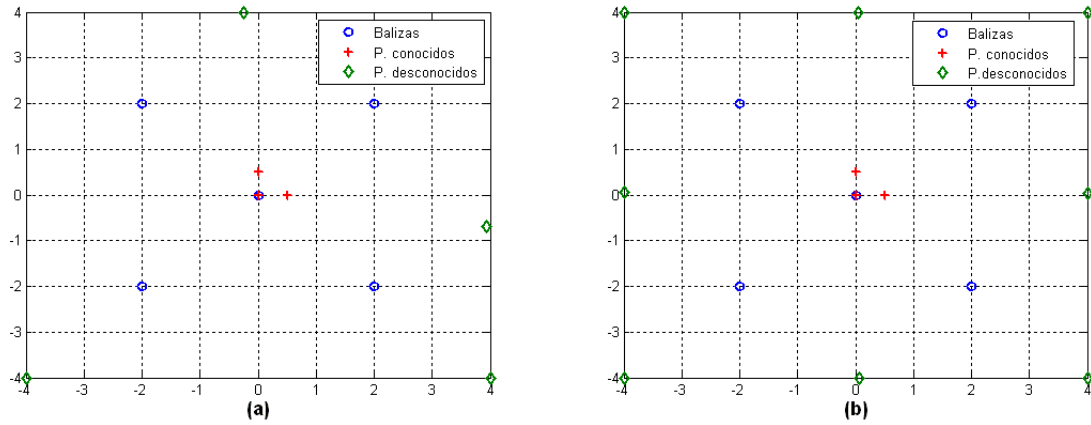


Figura 4.17: Representación de la ubicación de las zonas de puntos óptimos para la primera distribución de balizas en el plano $z=0$, para el caso hiperbólico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.

Tabla 4.12: Coordenadas de los puntos óptimos para la segunda distribución de balizas en el caso hiperbólico.

Posición de los puntos óptimos para la segunda distribución de balizas		
4 Puntos $[x, y, z]$ m.	8 Puntos $[x, y, z]$ m.	
$[4.0, -4.0, 0.0]$	$[-4.0, 0.5, 0.0]$	$[-4.0, 4.0, 0.0]$
$[-4.0, 2.7, 0.0]$	$[-2.2, -4.0, 0.0]$	$[1.0, 4.0, 0.0]$
$[-4.0, -2.0, 0.0]$	$[4.0, 4.0, 0.0]$	$[-4.0, 2.0, 0.0]$
$[1.5, 4.0, 0.0]$	$[4.0, -4.0, 0.0]$	$[-4.0, -2.2, 0.0]$

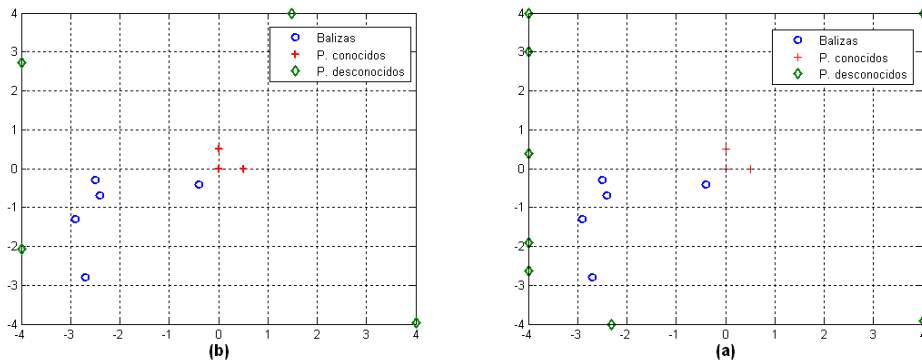


Figura 4.18: Representación de la ubicación de las zonas de puntos óptimos para la segunda distribución de balizas en el plano $z=0$, para el caso hiperbólico. (a) Se buscan 4 puntos óptimos. (b) Se buscan 8 puntos óptimos.

4.4.2.2. Puntos óptimos Vs. puntos aleatorios

Al igual que en el caso esférico se van a comparar los resultados obtenidos situando los puntos de test de forma aleatoria o ubicándolos en las zonas óptimas. Las condiciones de las pruebas son las mismas que en el caso esférico exceptuando que los valores de desviación típica de ruido simulados en las medidas, en este caso $\sigma = \{0.001, 0.01, 0.015\}$ m.

Para valores superiores, el algoritmo de autocalibración situando los puntos de test desconocidos de forma aleatoria no converge.

Para la primera distribución de balizas la posición de los puntos de test desconocidos tanto aleatorios como óptimos queda reflejada en la tabla 4.13, (en el caso de los puntos óptimos son los calculados en el apartado correspondiente).

Tabla 4.13: Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la primera distribución de balizas y empleando un posicionamiento hiperbólico.

Posición de los puntos de test desconocidos en la primera prueba	
4 Puntos aleatorios $[x, y, z]$ m.	4 Puntos óptimos $[x, y, z]$ m.
$[-2.9, -2.4, 0.0]$	$[-0.25, 4.0, 0.0]$
$[-2.4, 0.8, 0.0]$	$[4.0, -4.0, 0.0]$
$[-1.8, -2.4, 0.0]$	$[4.0, -0.69, 0.0]$
$[-3.9, 2.0, 0.0]$	$[-4.0, -4.0, 0.0]$

En la figura 4.20 se muestran los resultados de la autocalibración para las cien simulaciones con la primera distribución de balizas. En este caso la dispersión en los resultados se ve muy reducida si se sitúan los puntos de test desconocidos en las zonas óptimas.

En el gráfico de la figura 4.19 se puede observar la desviación típica de las 100 simulaciones en cada una de las componentes $[x, y, z]$ para los diferentes niveles de ruido. Se puede ver cómo, al aumentar el nivel de ruido en las medidas, aumenta la desviación típica en los resultados y cómo el empleo de los puntos óptimos reduce esta dispersión en la mayoría de los casos a más de un 50%, consiguiendo de esta forma que el error medio en la estimación de las balizas sea mucho menor, (ver tabla 4.14).

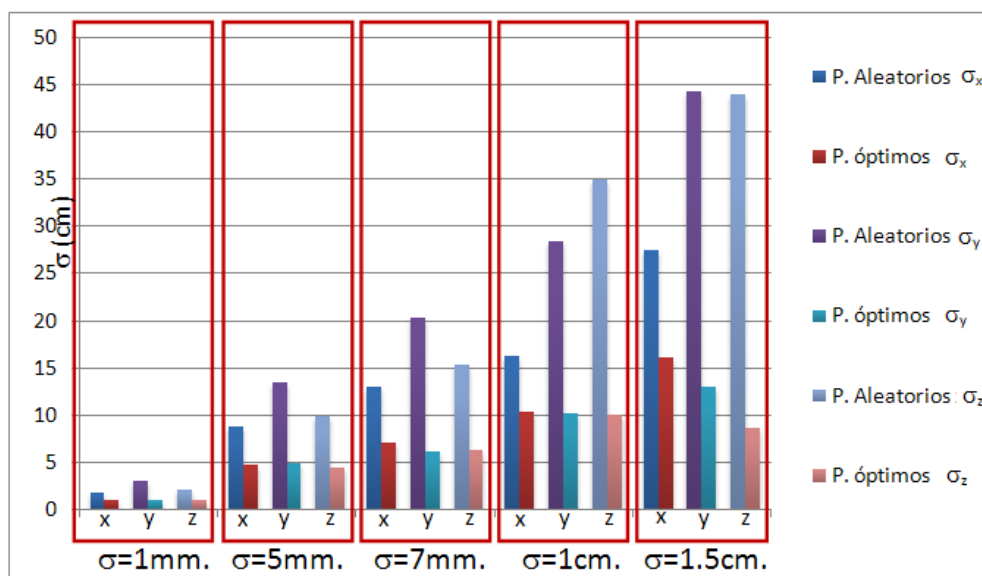


Figura 4.19: Comparación de la desviación típica en cada una de las componentes (x, y, z) , tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la primera distribución de balizas con posicionamiento hiperbólico.

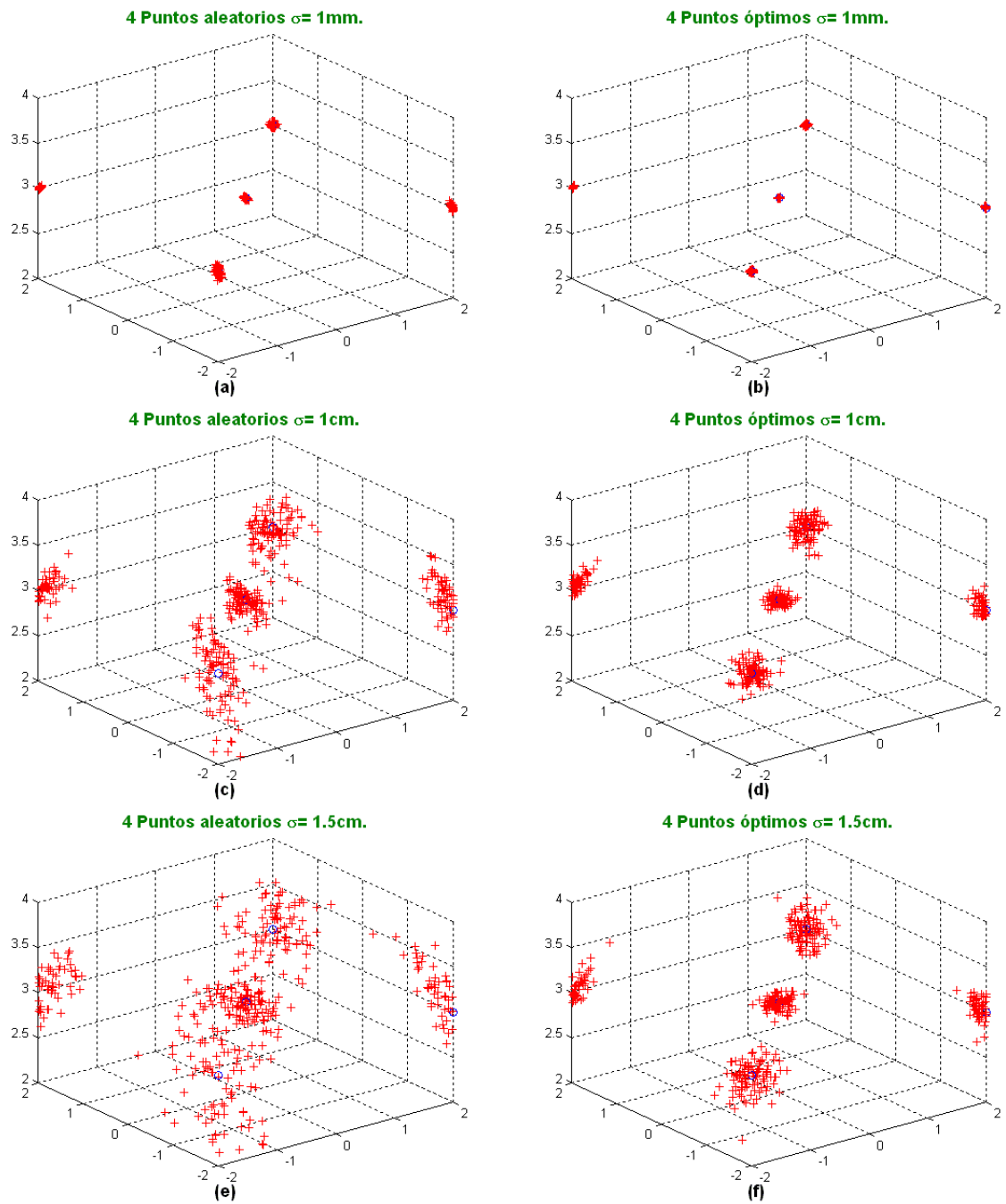


Figura 4.20: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\}$ cm situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la primera distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.

Tabla 4.14: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), empleando puntos aleatorios y óptimos con la primera distribución de balizas y posicionamiento hiperbólico.

	Puntos Aleatorios (cm)			Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.16	0.33	0.19	0.02	0.04	0.05
$\sigma = 5 \text{ mm.}$	0.47	1.04	0.99	0.30	0.63	0.32
$\sigma = 7 \text{ mm.}$	0.73	1.35	1.65	0.44	0.90	0.39
$\sigma = 1 \text{ cm.}$	1.08	1.86	2.46	0.71	1.09	0.42
$\sigma = 1.5 \text{ cm.}$	1.72	2.79	3.93	0.90	1.31	0.75

En la tabla 4.15 se muestra la posición de los puntos de test desconocidos para la segunda distribución de balizas tanto si se posicionan de forma aleatoria u óptima

Tabla 4.15: Ubicación de los puntos de test desconocidos tanto si se colocan de forma aleatoria como óptima para la segunda distribución de balizas y empleando un posicionamiento hiperbólico.

Posición de los puntos de test desconocidos en la segunda prueba	
4 Puntos aleatorios $[x, y, z]$ m.	4 Puntos óptimos $[x, y, z]$ m.
$[2.5, 3.2, 0.0]$	$[4.0, -4.0, 0.0]$
$[-2.9, 3.3, 0.0]$	$[-4.0, 2.7, 0.0]$
$[1.0, -3.2, 0.0]$	$[-4.0, -2.0, 0.0]$
$[-1.7, 0.3, 0.0]$	$[1.5, 4.0, 0.0]$

En la figura 4.21 se pueden observar los resultados de la prueba para la segunda distribución de balizas. En esta ocasión, al haberse seleccionado una mala distribución de balizas, se observa que la dispersión de los resultados es muy elevada. De hecho, no se puede distinguir a qué baliza corresponde cada punto para desviaciones típicas de ruido en las medidas superiores a 1 mm.

La figura 4.22 muestra los valores de desviación de la prueba realizada. Se puede observar que en este caso los valores son muy superiores al del resto de las pruebas, esto hace que los errores medios, (ver tabla 4.16), sean muy altos, cercanos al medio metro en alguno de los casos si se distribuyen los puntos desconocidos de forma aleatoria.

También se puede ver que para este tipo de distribuciones es muy necesario el empleo de puntos óptimos, sobre todo para niveles de ruido en las medidas superiores a 1 cm, ya que si se emplean estos puntos se evita que se dispare el error en la estimación de las balizas.

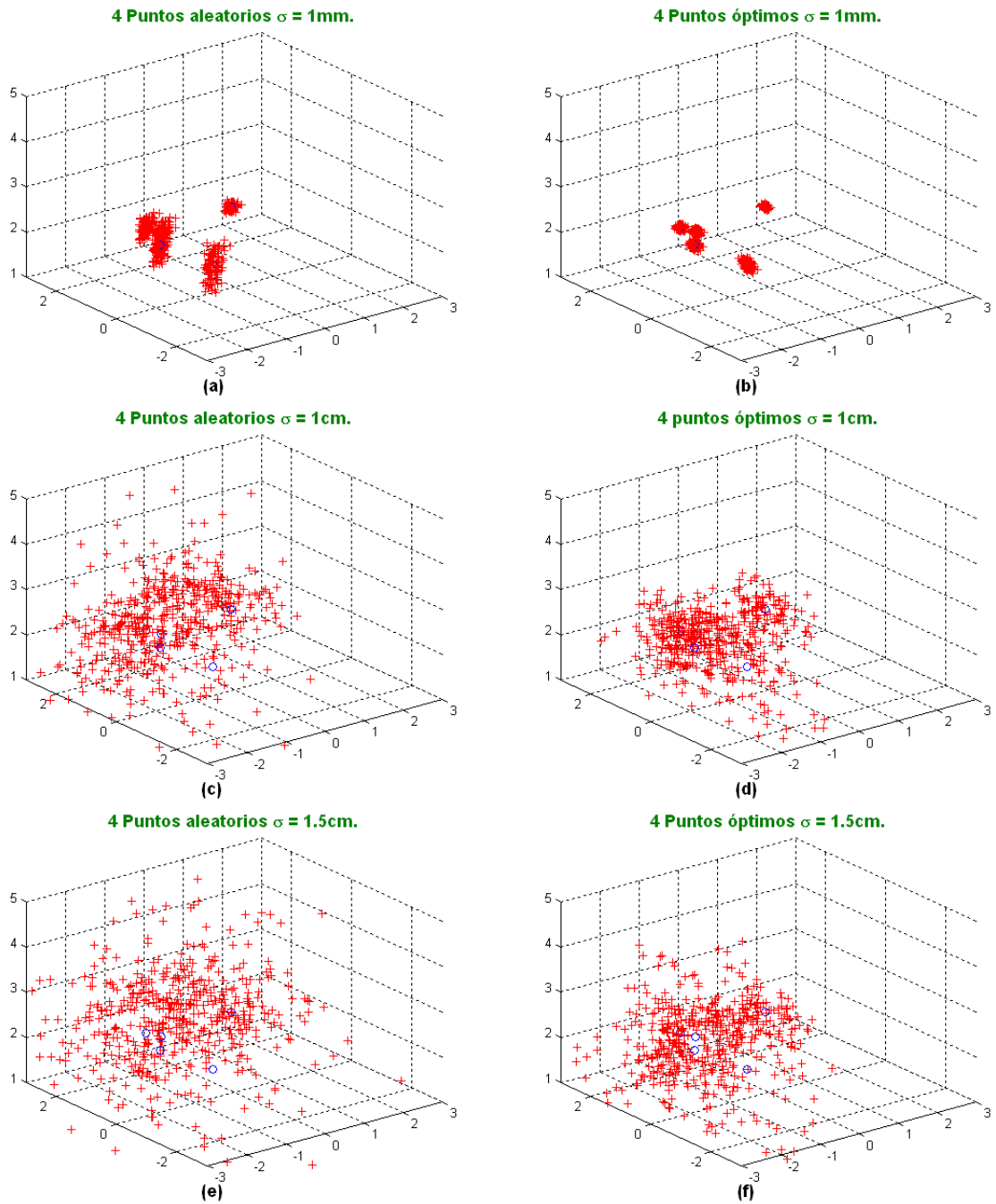


Figura 4.21: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\} \text{ cm}$ situando los puntos de test desconocidos de forma aleatoria/óptima y empleando la segunda distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) Puntos de medida desconocidos situados de forma aleatoria. (b), (d) y (f) Los puntos de test desconocidos se sitúan en las zonas óptimas.

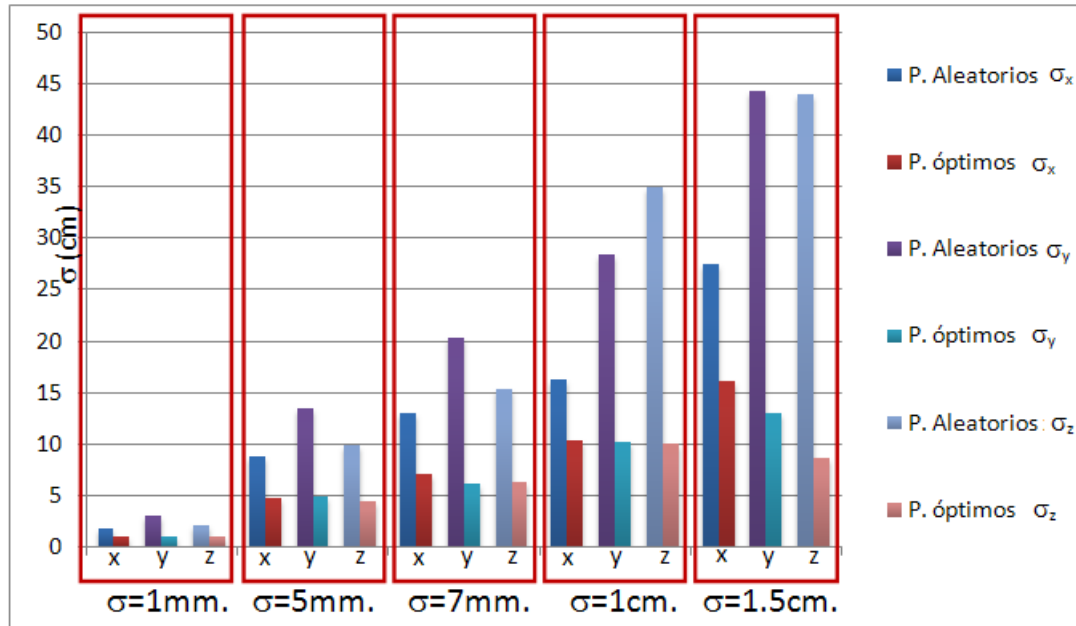


Figura 4.22: Comparación de la desviación típica en cada una de las componentes (x, y, z), tanto si se sitúan los puntos de medida de forma aleatoria u óptima para la segunda distribución de balizas con posicionamiento hiperbólico.

Tabla 4.16: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), empleando puntos aleatorios y óptimos con la segunda distribución de balizas y posicionamiento hiperbólico.

	Puntos Aleatorios (cm)			Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.49	1.42	0.89	0.22	0.31	0.61
$\sigma = 5 \text{ mm.}$	4.81	16.03	4.37	3.82	5.86	3.69
$\sigma = 7 \text{ mm.}$	8.72	15.99	5.24	4.82	6.61	5.11
$\sigma = 1 \text{ cm.}$	10.17	37.89	12.33	6.54	7.52	6.73
$\sigma = 1.5 \text{ cm.}$	43.01	46.70	22.21	8.36	9.28	8.02

4.4.2.3. Número de puntos de test

En esta sección se comprueba si el aumento del número de puntos de test desconocidos consigue disminuir el error que se comete al realizar la autocalibración. Para ello se realizan una serie de tests empleando 4 y 8 puntos de medida desconocidos y se comparan los resultados. Se emplean los puntos óptimos obtenidos en las tablas 4.11 y 4.12.

En la figura 4.23 se muestra el resultado de las 100 autocalibraciones y se puede apreciar cómo la dispersión en las soluciones se reduce en el caso de emplear los 8 puntos de medida. Esto queda constatado en la figura 4.24 donde se observa que la desviación típica se reduce aproximadamente en un 25 % en el caso de emplear 8 puntos de test desconocidos.

La mejora en los resultados también se refleja en la tabla 4.17 donde se observa la disminución de los errores medios en la ubicación de las balizas al aumentar el número

de puntos de test desconocidos, se puede observar una mejora muy importante sobre todo para los niveles de ruido más elevados $\sigma = \{1, 1.5\} \text{ cm}$ donde los errores medios se reducen en más de un 60 %.

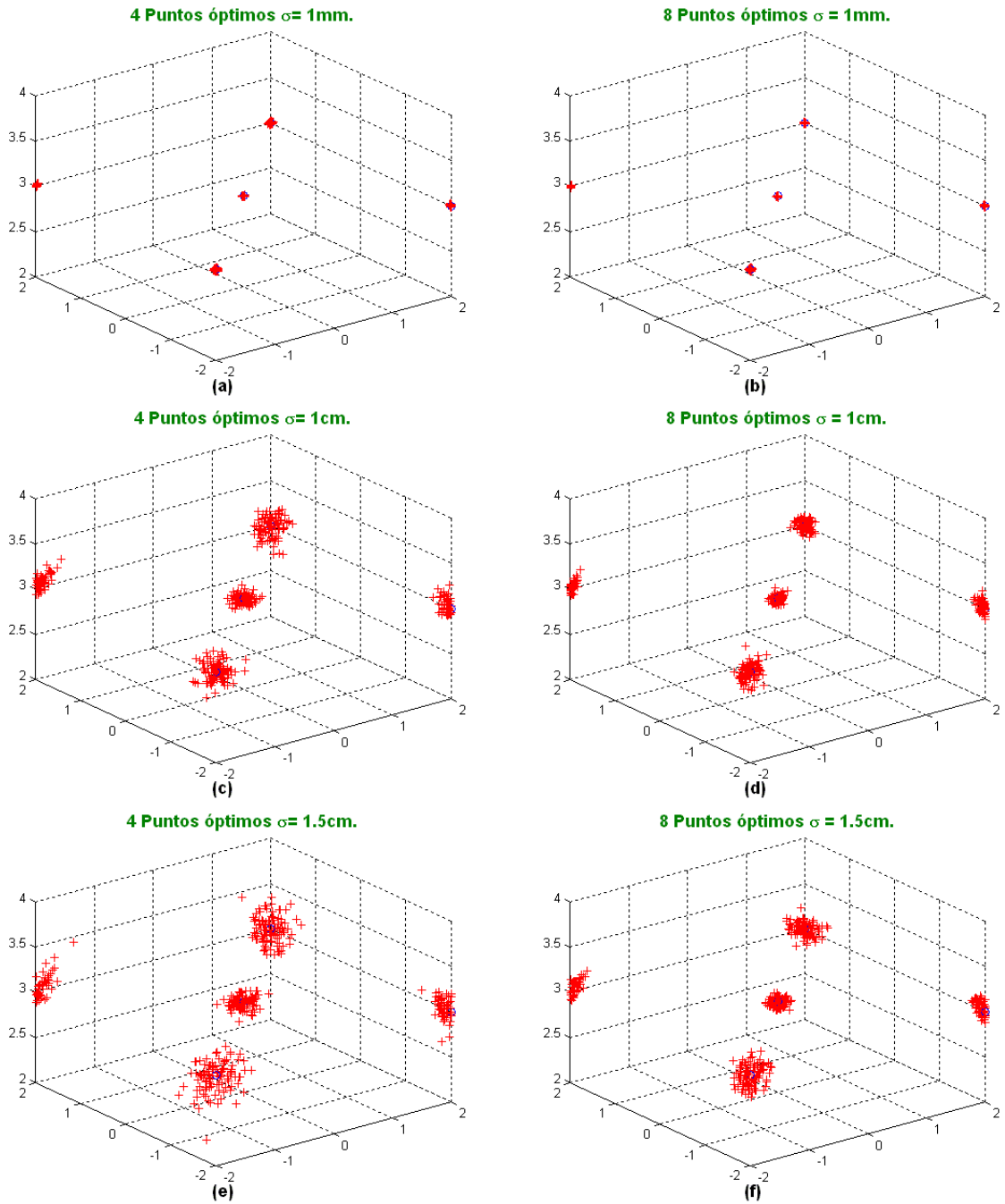


Figura 4.23: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\} \text{ cm}$ situando los puntos de test desconocidos de forma óptima y empleando la primera distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima.

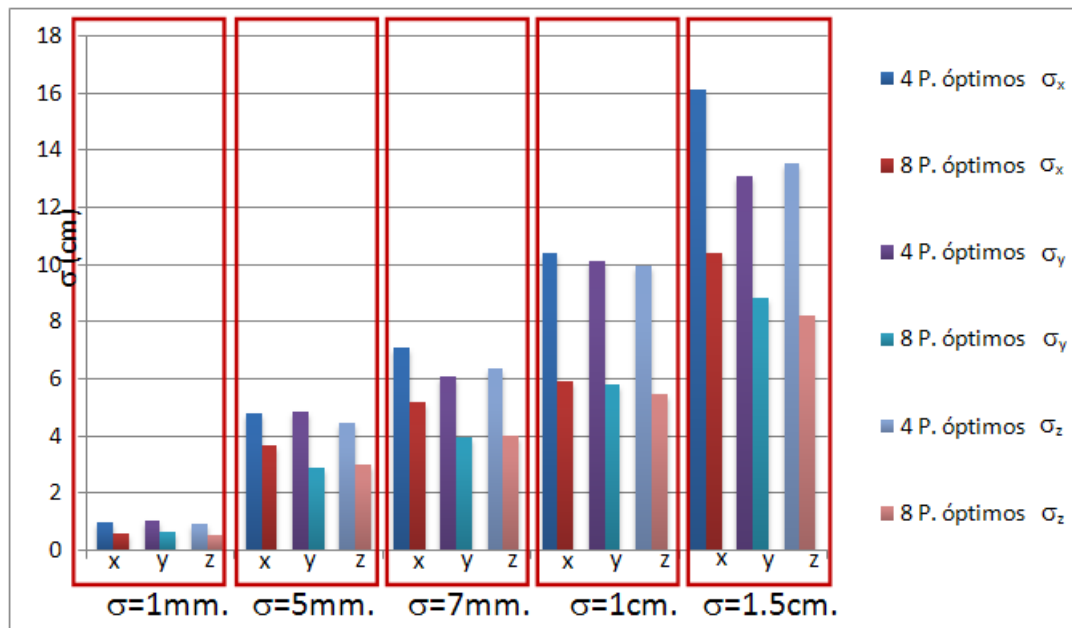


Figura 4.24: Comparación de la desviación típica en cada una de las componentes (x, y, z) , aumentando los puntos de medida, para la primera distribución de balizas con posicionamiento hiperbólico.

Tabla 4.17: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), variando el número de puntos de test con la primera distribución de balizas y posicionamiento hiperbólico.

	4 Puntos Óptimos (cm)			8 Puntos Óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.02	0.04	0.05	0.01	0.02	0.02
$\sigma = 5 \text{ mm.}$	0.30	0.63	0.32	0.27	0.18	0.17
$\sigma = 7 \text{ mm.}$	0.44	0.90	0.39	0.29	0.24	0.23
$\sigma = 1 \text{ cm.}$	0.71	1.09	0.42	0.31	0.29	0.26
$\sigma = 1.5 \text{ cm.}$	0.90	1.31	0.75	0.38	0.31	0.30

Por último se efectúa la misma prueba para la segunda distribución de balizas. En la figura 4.25 se muestran los resultados de las autocalibraciones. En este caso se puede ver cómo la dispersión en los resultados es bastante parecida en ambos casos.

En el gráfico de la figura 4.26 se puede comprobar que la desviación típica de los resultados en cada componente es bastante parecida tanto si se emplean 4 como 8 puntos de test desconocidos. Hay una leve mejoría con 8 puntos de test pero es muy pequeña, esto hace que los errores medios cometidos en la ubicación de las balizas sean muy parecidos. Para la mayoría de los casos la mejora está en torno a 1cm (ver tabla 4.18).

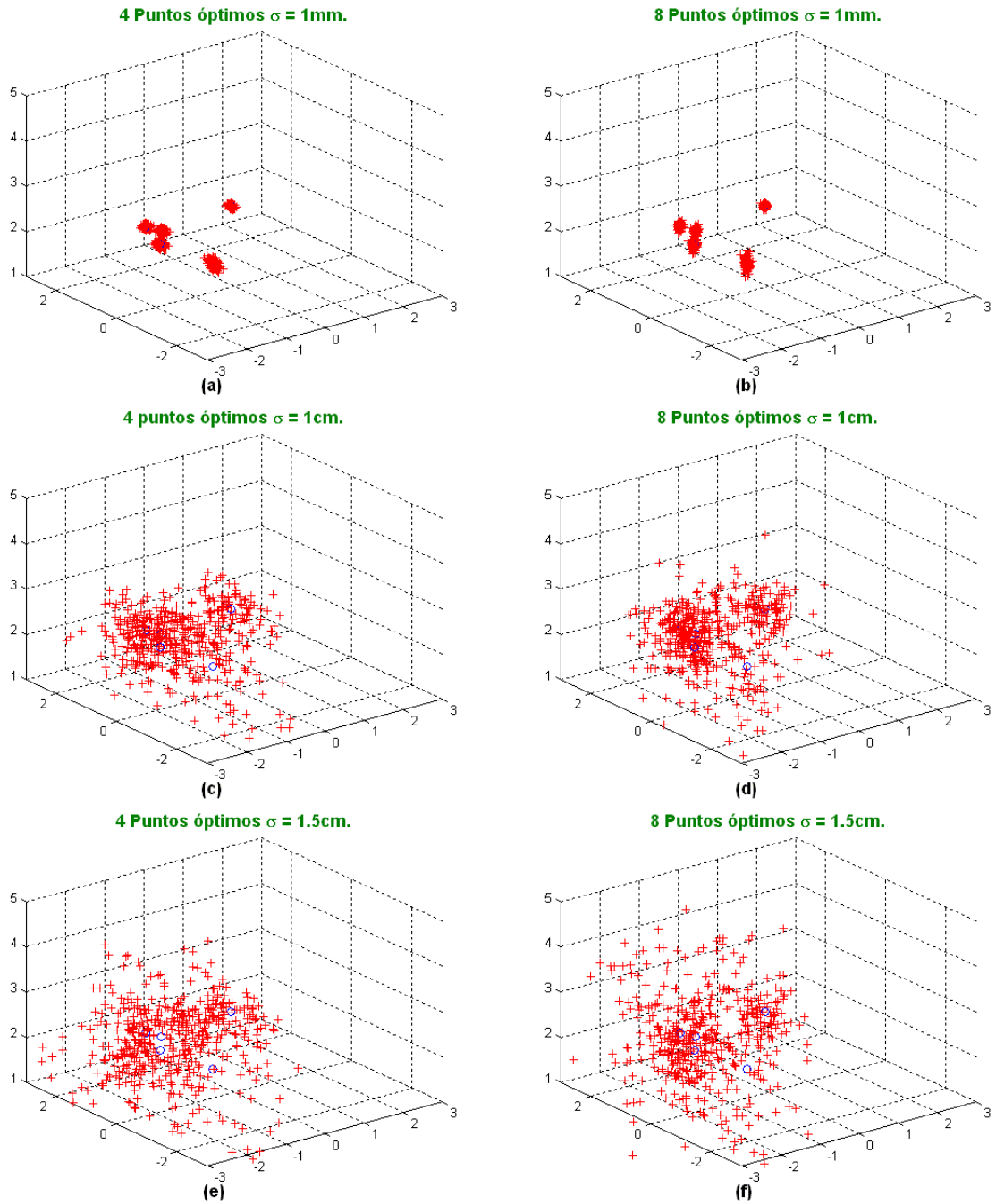


Figura 4.25: Representación de las 100 autocalibraciones para diferentes niveles de ruido en las medidas $\sigma = \{0.1, 1, 1.5\} \text{ cm}$ situando los puntos de test desconocidos de forma óptima y empleando la segunda distribución de balizas con posicionamiento hiperbólico. (a), (c) y (e) 4 Puntos de medida desconocidos situados de forma óptima. (b), (d) y (f) 8 Puntos de medida desconocidos situados de forma óptima.

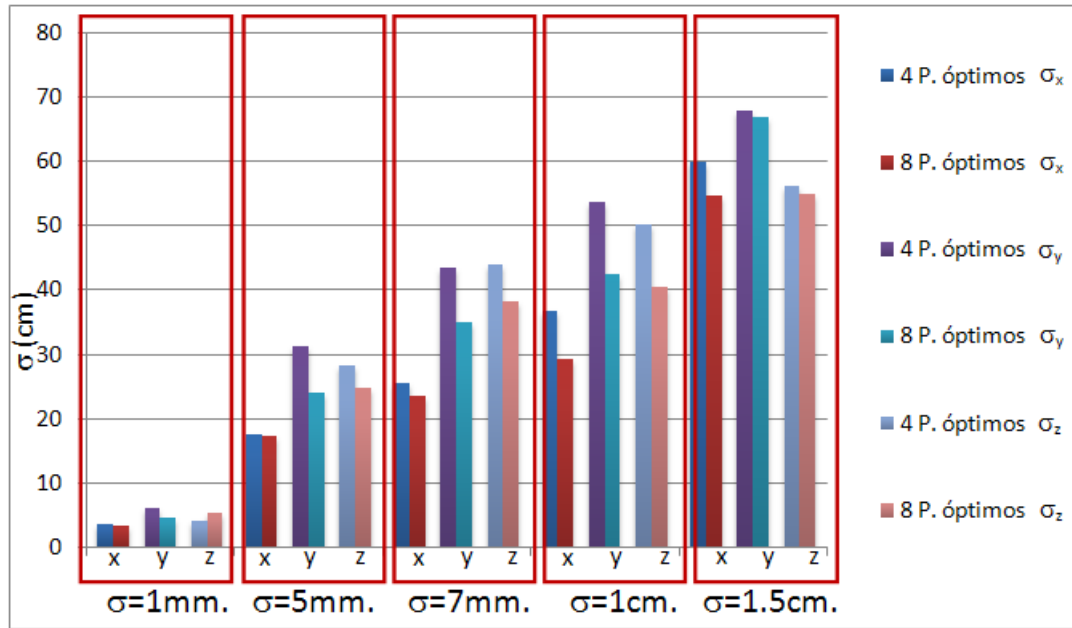


Figura 4.26: Comparación de la desviación típica en cada una de las componentes (x, y, z) , aumentando los puntos de medida, para la segunda distribución de balizas con posicionamiento hiperbólico.

Tabla 4.18: Error medio en la localización de las balizas por componentes $[x, y, z]$ (cm), variando el número de puntos de test con la segunda distribución de balizas y posicionamiento hiperbólico.

	4 Puntos óptimos (cm)			8 puntos óptimos (cm)		
	x	y	z	x	y	z
$\sigma = 1 \text{ mm.}$	0.22	0.31	0.61	0.1	0.29	0.29
$\sigma = 5 \text{ mm.}$	3.82	5.86	3.69	2.85	3.45	2.49
$\sigma = 7 \text{ mm.}$	4.82	6.61	5.11	4.64	3.37	3.05
$\sigma = 1 \text{ cm.}$	6.54	7.52	6.73	4.94	6.36	3.32
$\sigma = 1.5 \text{ cm.}$	8.36	9.28	8.02	7.31	8.74	7.61

4.5. Conclusiones

En este capítulo se ha presentado un método para realizar la calibración de un sistema LPS balizado tanto si se emplea posicionamiento esférico como hiperbólico. El algoritmo propuesto necesita tomar diversas medidas en varios puntos de la zona de cobertura para después encontrar las posiciones de las balizas empleando el método de resolución directa de las ecuaciones de distancia mediante el algoritmo de Gauss-Newton. De esos puntos de medida sólo se necesita conocer la ubicación de tres puntos de medida, mientras que del resto no es necesario que se conozca su posición. Además, también se propone un método que permite obtener las mejores zonas para ubicar los puntos de test desconocidos. De las pruebas realizadas bajo simulación se pueden extraer varias conclusiones:

- En el caso esférico, se ha visto en las pruebas bajo simulación, que los resultados obtenidos para una correcta localización de balizas, (esto es bajo PDOP en el área de cobertura) mejoran ligeramente ubicando los puntos de medida en las zonas óptimas, sobre todo para desviaciones típicas de ruido en las medidas superiores a 3 cm. Esta mejora es más notable en el caso de que la distribución de balizas presente un elevado PDOP (bajo PDOP en el área de cobertura).
- Además, se ha comprobado que en el caso esférico el aumento de los puntos de medida desconocidos en las zonas óptimas consigue que mejoren los resultados en las autocalibraciones.
- Para el caso hiperbólico se ha comprobado que, si se quiere realizar una autocalibración, es muy recomendable ubicar los puntos de medida en las zonas óptimas ya que si no los errores son demasiado elevados. Por ejemplo: para una distribución óptima de balizas y con una desviación típica de ruido en las medidas de 1 cm, el error puede reducirse en torno al 30 %, mientras que en las pruebas con una distribución de balizas con un elevado PDOP, el error en la autocalibración puede reducirse en torno al 50 %.
- Se han ejecutado simulaciones para el caso hiperbólico aumentando el número de puntos de medida. Se ha comprobado que en el caso de que la distribución de balizas tenga un elevado PDOP, apenas se produce mejora, mientras que si la distribución de balizas es buena, (esto es bajo PDOP), sí que se consigue una mejora en los resultados con reducción de errores de hasta el 60 %.
- A la hora de comparar los resultados obtenidos en el caso esférico y en el caso hiperbólico, se ha comprobado que los resultados obtenidos si se emplea un posicionamiento esférico son mucho mejores que si se emplea un posicionamiento hiperbólico. En las simulaciones se ha comprobado que para una buena distribución de balizas (esto es que tengan el menor PDOP posible en la zona de cobertura) el error medio cometido en el primer caso está en torno a un centímetro para desviaciones típicas de ruido en las medidas de 4 cm, mientras que en el segundo caso sube hasta casi los 3 cm en el caso hiperbólico aún cuando se suponen desviaciones típicas de ruido en las medidas de 7 mm.

Calibración global de varios LPS ultrasónicos mediante un robot móvil.

En el capítulo anterior se ha propuesto un mecanismo de autocalibración para un sistema de posicionamiento local individual. En este capítulo se propone un método para calibrar varios sistemas de posicionamiento independientes de forma global, esto es que todos los sistemas compartan un mismo origen de referencia.

Para la realización de la autocalibración de los diferentes sistemas de posicionamiento se emplea un robot móvil, ya que el método propuesto fusiona, mediante un filtro $H-\infty$, los datos proporcionados por la odometría del robot (Velocidad Lineal y Velocidad Angular) con la información de distancia proporcionada por cada LPS, y mediante un proceso iterativo calcula tanto la posición del robot como la ubicación de las balizas de los LPS. Además, el método propuesto utiliza como origen de referencia del sistema global la posición inicial del robot.

El capítulo se ha estructurado de la siguiente manera: en el apartado 5.1 se realiza una descripción del método de calibración propuesto para cada uno de los LPS individuales y se plantea cómo se utiliza el filtro $H-\infty$ para la obtención de la posición de las balizas; a continuación en el punto 5.2 se analiza el método propuesto para extenderlo a varios LPS y se propone un algoritmo para minimizar los errores debidos a las zonas donde el robot navega sin apoyo de ningún LPS; seguidamente en el apartado 5.3 se presentan varias pruebas realizadas en un entorno de simulación para comprobar las propiedades de los métodos propuestos; por último en el apartado 5.4 se presentan las conclusiones extraídas de este capítulo.

5.1. Calibración para cada LPS individual

Una vez el robot llegue al área de cobertura de un LPS se realiza el proceso de calibración del mismo, este proceso se divide en tres etapas (figura 5.1):

1. **Fase de Precalibración.** Es la fase inicial del algoritmo, se realiza un pequeño recorrido dentro de la zona de cobertura para obtener una estimación inicial de la posición de las balizas.
2. **Fase de Calibración.** Se emplea tanto la información de odometría del robot, como la información proporcionada por el LPS para realizar tanto la calibración de las balizas del sistema LPS como para obtener la posición del robot. Este proceso se realiza mediante un filtro $H-\infty$.
3. **Fase de Navegación.** Una vez finalizado el proceso de calibración se fija la posición de las balizas dentro del sistema y se emplea el filtro $H-\infty$ para guiar al robot dentro del área de cobertura.

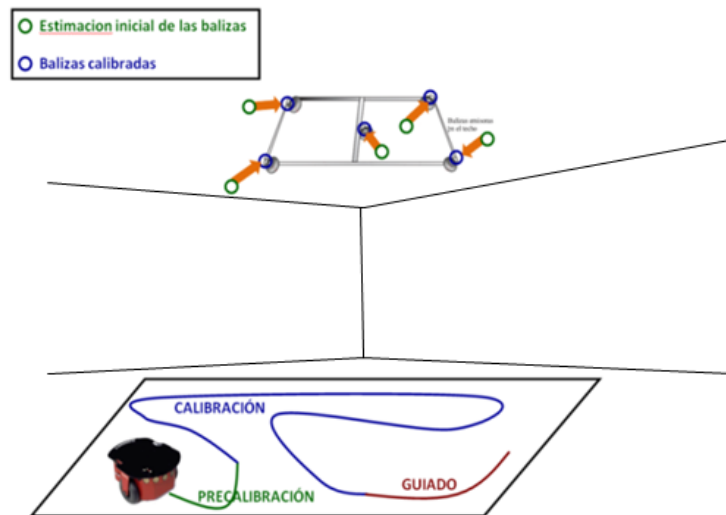


Figura 5.1: Fases propuestas para la calibración de un LPS

5.1.1. Fase de Precalibración

Esta fase se lleva a cabo porque para realizar el proceso de calibración se necesita conocer una estimación de la posición inicial de la ubicación de las balizas. Para ello se realiza un pequeño recorrido dentro del área de cobertura y se realiza un posicionamiento inverso para ubicar las balizas, esto es, se almacena la posición del robot y las distancias a cada una de las balizas y se emplean esos datos para localizar cada una de las balizas del sistema. Para realizar el posicionamiento inverso y calcular la posición de las balizas se emplea el algoritmo de posicionamiento basado en las propiedades de los bideterminantes de Cayley-Menger. Para ello se toman los puntos realizados en el recorrido como si fueran las balizas y se calcula con las distancias medidas la posición de las balizas (figura 5.2) .

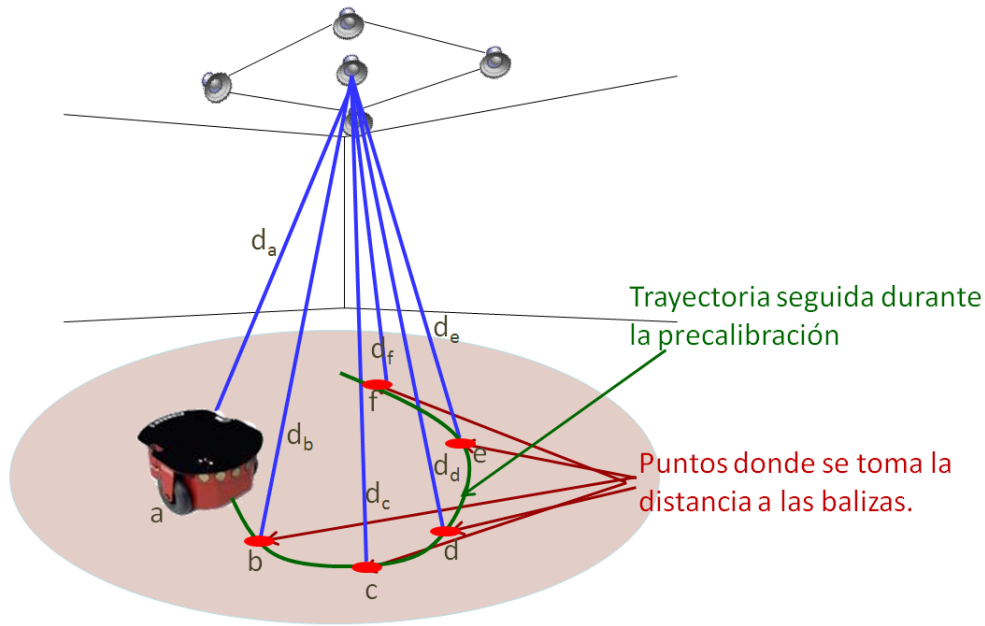


Figura 5.2: Ejemplo de obtención de la posición de una baliza a partir de las distancias medida entre el móvil y la baliza en varios puntos de la trayectoria de precalibración.

5.1.2. Fase de Calibración

Uno de los mecanismos más usados para la navegación empleando robots es el filtro de Kalman, también conocido como filtro H_2 [ATSW04, MLC⁺06, LHAR99]. Este filtro es una herramienta muy eficiente en la estimación de estados de un sistema; una de sus principales características es que minimiza la varianza del error de estimación suponiendo un ruido gaussiano de media cero.

Pero el Filtro de Kalman tiene una serie de limitaciones: su comportamiento no es óptimo cuando la componente continua de ruido no es nula; tampoco es óptimo cuando se desconoce total o parcialmente la estadística de ruido del sistema.

Debido a las características del sistema propuesto, en esta tesis en vez del filtro de Kalman se propone emplear como método de fusión de datos el filtro $H-\infty$, también conocido como filtro *minimax* [SD97, Sim05], que se caracteriza por minimizar el máximo error de estimación. Además en [ST92] se demuestra como el filtro $H-\infty$ es menos sensible a las variaciones de parámetros que su homólogo filtro de Kalman. Esto, en definitiva, permite que se pueda emplear el filtro $H-\infty$ aunque no se tenga un modelo plenamente fiable del sistema.

El empleo de este filtro es muy similar al del filtro de Kalman en navegación de robots móviles; se actualiza un vector de estado de la siguiente forma:

1. Etapa de predicción. Se emplea la información de odometría del robot para estimar el estado.

2. Etapa de actualización. Se actualiza el vector de estado con las medidas externas provenientes del LPS.

En este caso el vector de estado, \mathbf{q}_k , está compuesto por la posición del robot, su orientación y la posición de las balizas.

5.1.2.1. Formulación del filtro H_∞

Si se define \mathbf{q}_k como el vector de estado del sistema en un instante k , para poder emplear el filtro H_∞ es necesario resolver la siguiente ecuación [SD97]:

$$\min_{\hat{\mathbf{q}}_k} \max_{\mathbf{w}_k, \mathbf{v}_k} (J) \quad (5.1)$$

Donde J representa un índice de comportamiento del filtro; $\hat{\mathbf{q}}_k$ es la estimación del vector de estado realizada; \mathbf{w}_k y \mathbf{v}_k son las componentes de ruido. La estimación $\hat{\mathbf{q}}_k$ en el instante $k = N$ es obtenida basándose en la información disponible hasta $N-1$.

Dados los peores valores de \mathbf{w}_k y \mathbf{v}_k , el filtro pretende realizar una estimación que minimice el peor efecto que las componentes de ruido tienen sobre el error de estimación. La función J se define como sigue [SD97]:

$$J = \frac{\sum_{k=0}^{N-1} \|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_{\mathbf{Q}}^2}{\sum_{k=0}^{N-1} \|\mathbf{w}_k\|_{\mathbf{W}}^2 + \sum_{k=0}^{N-1} \|\mathbf{v}_k\|_{\mathbf{V}}^2} \quad (5.2)$$

Donde $\mathbf{W} > 0$, $\mathbf{V} > 0$, $\mathbf{Q} > 0$ son matrices de ponderación, calculándose las normas ponderadas de los vectores como $\|\mathbf{s}\|_{\mathbf{H}}^2 = \mathbf{s}^T \mathbf{H} \cdot \mathbf{s}$, siendo \mathbf{s} un vector genérico y \mathbf{H} una matriz de ponderación. Por una parte se desea que la estimación minimice el valor de J , debiendo por tanto estar $\hat{\mathbf{q}}_k$ lo más próximo posible a \mathbf{q}_k . Por otro lado, las componentes de ruido podrían ser tan grandes que la estimación no fuera demasiado buena, pero este efecto se ve compensado al aparecer las normas de los vectores de ruido ponderadas por \mathbf{W} y \mathbf{V} en el denominador de la expresión 5.2. Las matrices de ponderación \mathbf{W} , \mathbf{V} y \mathbf{Q} son elegidas por el diseñador [SFMdSPC01].

Analizando lo anterior, se puede observar que el problema de la estimación es complejo de resolver matemáticamente. Sin embargo, se puede establecer que la estimación óptima de \mathbf{q}_k entre todas las posibles estimaciones de $\hat{\mathbf{q}}_k$ satisfaga la condición:

$$J < 1/\gamma \quad (5.3)$$

Donde $\gamma > 0$ es un escalar elegido por el diseñador. Este valor puede considerarse como un nivel predeterminado de atenuación en el ruido.

Con todas estas consideraciones la ecuación del filtro H- ∞ , [SD97], adaptadas al problema aquí planteado, puede expresarse como sigue:

- Etapa de predicción:

$$\hat{\mathbf{q}}_{k,odo} = f(\hat{\mathbf{q}}_{k-1}, sistema)$$

- Etapa de corrección:

$$\begin{aligned} \mathbf{L}_k &= (\mathbf{I} - \gamma \cdot \mathbf{Q}_k \cdot \mathbf{P}_k + \mathbf{C}_k^T \cdot \mathbf{V}^{-1} \cdot \mathbf{C}_k \cdot \mathbf{P}_k)^{-1} \\ \mathbf{K}_k &= \mathbf{A}_k \cdot \mathbf{P}_k \cdot \mathbf{L}_k \cdot \mathbf{C}_k^T \cdot \mathbf{V}^{-1} \\ \hat{\mathbf{q}}_k &= \hat{\mathbf{q}}_{k,odo} + \mathbf{K}_k (\mathbf{d}_k - \hat{\mathbf{d}}_k) \\ \mathbf{P}_{k+1} &= \mathbf{A}_k \cdot \mathbf{P}_k \cdot \mathbf{L}_k \cdot \mathbf{A}_k \cdot \mathbf{W} \end{aligned} \quad (5.4)$$

Donde:

- $f(\hat{\mathbf{q}}_{k-1}, sistema)$ es la función que obtiene la posición del robot móvil en el instante de tiempo k a partir de la posición del robot en el instante $k - 1$ empleando los datos de odometría. Se define en la sección 5.1.2.2.
- $\hat{\mathbf{q}}_{k,odo}$ es el vector estimado que contiene la posición y la orientación del robot y la localización de las balizas teniendo en cuenta sólo los datos de odometría.
- $\hat{\mathbf{q}}_k$ es el vector que contiene las variables a estimar, en este caso, la localización y orientación del robot y la posición de las balizas, una vez realizada la corrección.
- \mathbf{K}_k es la ganancia del filtro.
- \mathbf{d}_k es el vector que contiene las variables medibles, en este caso las distancias medidas a las balizas.
- $\hat{\mathbf{d}}_k$ es el vector que contiene las medidas estimadas. En este caso la distancia desde las balizas estimadas a la posición estimada del móvil.
- \mathbf{A}_k y \mathbf{C}_k son las matrices Jacobianas de la dinámica del sistema. En este caso las del robot y el LPS respectivamente. Se definen posteriormente.
- \mathbf{V} y \mathbf{W} son matrices relacionadas con el ruido en el sistema, en este caso se relacionan con el ruido de la odometría y las medidas del LPS respectivamente.
- \mathbf{I} es la matriz identidad.
- \mathbf{P}_k es la matriz de covarianza del filtro.
- $\gamma \cdot \mathbf{Q}_k$ es un parámetro de ajuste del filtro.

5.1.2.2. Descripción de la odometría del robot

El primer paso consiste en definir el vector de estado del sistema, esto es, las incógnitas del mismo. Para un sistema LPS compuesto por N balizas de coordenadas (bx_i, by_i, bz_i) con $i = 1 \dots N$ y un robot móvil con un ángulo de orientación θ_k y que se encuentra en las coordenadas (x_k, y_k) (se supone conocida la altura), el vector queda de la forma:

$$\hat{\mathbf{q}}_{k-1} = [x_{k-1}, y_{k-1}, \theta_{k-1}, bx_1, \dots, bz_N]^T \quad (5.5)$$

La actualización de la posición del robot a través de los datos de odometría puede obtenerse como [PM07]:

$$\hat{\mathbf{q}}_{k,odo} = \begin{bmatrix} x_k \\ y_k \\ \theta_k \\ bx_1 \\ \vdots \\ bz_n \end{bmatrix} = f(\hat{\mathbf{q}}_{k-1}, sistema) = \begin{bmatrix} x_{k-1} + \Delta D_k \cdot \cos(\theta_k) \\ y_{k-1} + \Delta D_k \cdot \sin(\theta_k) \\ \theta_{k-1} + \theta_k \\ bx_1 \\ \vdots \\ bz_n \end{bmatrix} \quad (5.6)$$

Donde ΔD_k es la distancia recorrida por el robot entre el instante de tiempo $k-1$ y el instante de tiempo k , e $\Delta \theta_k$ es la variación de la orientación del robot entre los dos instantes de tiempo (figura 5.3). Se ha supuesto que se conoce la altura del robot (z) y que las balizas son fijas y permanecen en la misma posición a lo largo del tiempo.

A partir del modelo de odometría del robot móvil es posible calcular su modelo dinámico, (\mathbf{A}_k) , derivando $\hat{\mathbf{q}}_{k,odo}$ respecto cada uno de las variables del vector de estado:

$$\mathbf{A}_k = \frac{\partial \hat{\mathbf{q}}_{k,odo}}{\partial \mathbf{q}_k} \Big|_{\mathbf{q}_k = \hat{\mathbf{q}}_k} = \begin{bmatrix} 1 & 0 & -\Delta D_k \cdot \sin(\theta_k) & 0 & \dots & 0 \\ 0 & 1 & \Delta D_k \cdot \cos(\theta_k) & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \quad (5.7)$$

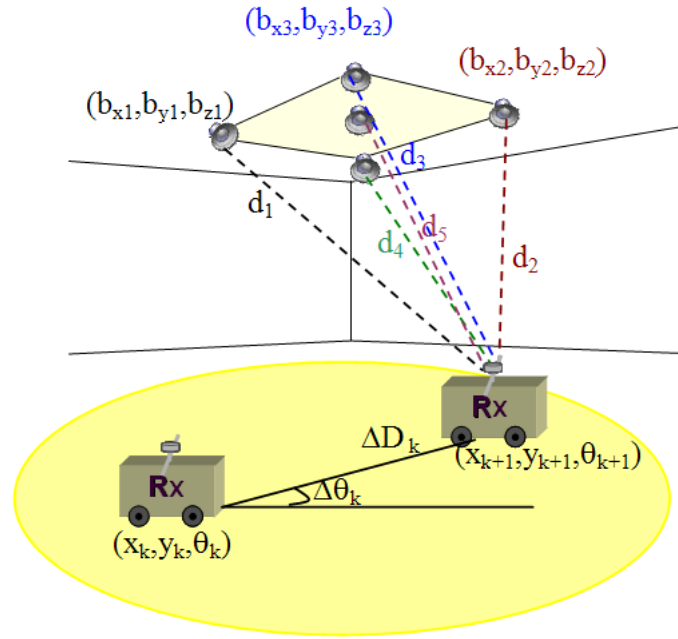


Figura 5.3: Variación de la posición del robot entre dos instantes de tiempo consecutivos.

5.1.2.3. Descripción del modelo del LPS

La distancia en el instante k entre la baliza i situada en (bx_i, by_i, bz_i) y el robot móvil situado en la posición (x_k, y_k, z_k) se puede expresar como:

$$d_{i,k} = \sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2} \quad (5.8)$$

Entonces el vector de distancias en el instante de tiempo k , entre cada una de las balizas que componen el LPS y el robot móvil puede expresarse como:

$$\mathbf{d}_k = \begin{bmatrix} d_{1,k} & d_{2,k} & \cdots & d_{N,k} \end{bmatrix} \quad (5.9)$$

Al igual que en el caso anterior, derivando 5.9 respecto a las variables del sistema, se obtiene el modelo dinámico de la distancia (\mathbf{C}_k):

$$\mathbf{C}_k = \frac{\partial \mathbf{d}_k}{\partial \mathbf{q}_k | \mathbf{q}_k = \hat{\mathbf{q}}_k} = \begin{bmatrix} \frac{\partial d_{1,k}}{\partial x_k} & \cdots & \frac{\partial d_{1,k}}{\partial \theta_k} & \frac{\partial d_{1,k}}{\partial bx_1} & \cdots & \frac{\partial d_{1,k}}{\partial bz_N} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial d_{N,k}}{\partial x_k} & \cdots & \frac{\partial d_{N,k}}{\partial \theta_k} & \frac{\partial d_{N,k}}{\partial bx_1} & \cdots & \frac{\partial d_{N,k}}{\partial bz_N} \end{bmatrix}$$

Donde:

$$\begin{aligned}
\frac{\partial d_{i,k}}{\partial x_k} &= \frac{(x_k - bx_i)}{\sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2}} \\
\frac{\partial d_{i,k}}{\partial y_k} &= \frac{(y_k - by_i)}{\sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2}} \\
\frac{\partial d_{i,k}}{\partial z_k} &= \frac{(z_k - bz_i)}{\sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2}} \\
\frac{\partial d_{i,k}}{\partial \theta_k} &= 0 \\
\frac{\partial d_{i,k}}{\partial bx_m} &= \frac{\partial d_{i,k}}{\partial by_m} = \frac{\partial d_{i,k}}{\partial bz_m} = 0 \quad m \neq i \\
\frac{\partial d_{i,k}}{\partial bx_i} &= \frac{-(x_k - bx_i)}{\sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2}} \\
\frac{\partial d_{i,k}}{\partial by_i} &= \frac{-(y_k - by_i)}{\sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2}} \\
\frac{\partial d_{i,k}}{\partial bz_i} &= \frac{-(z_k - bz_i)}{\sqrt{(x_k - bx_i)^2 + (y_k - by_i)^2 + (z_k - bz_i)^2}}
\end{aligned} \tag{5.10}$$

5.1.2.4. Modelado de las fuentes de ruido en el sistema

En el sistema propuesto se va a suponer que las principales fuentes de ruido son la odometría y la medida de distancia desde el robot móvil a las balizas.

En el caso de la odometría se va a emplear el modelo de Chenavier y Crowley [CC92], referenciado en [Kir02]. Se asume que los errores son incorrelados y se modelan como:

$$\mathbf{V}_k = \begin{bmatrix} K_{ss} |\Delta D_k \cos(\theta_k)| & 0 & 0 \\ 0 & K_{ss} |\Delta D_k \sin(\theta_k)| & 0 \\ 0 & 0 & K_{s\theta} |\Delta D_k| + K_{\theta\theta} |\Delta \theta_k| \end{bmatrix} \tag{5.11}$$

Donde K_{ss} representa el coeficiente de deriva en la odometría respecto al camino recorrido durante los dos instantes de muestreo, $K_{s\theta}$ representa la deriva en ángulo cuando avanza el robot entre dos instantes de muestreo y $K_{\theta\theta}$ representa la deriva en el ángulo cuando el robot gira durante dos instantes de muestreo. El valor de estos coeficientes se ha establecido de forma experimental en $1e-4$ mm.

En el caso de las medidas de distancia se asume que los errores son gaussianos e in-correlados, por eso la matriz \mathbf{W} es una matriz diagonal cuyos elementos son la desviación típica del error de medida de distancia:

$$\mathbf{W}_k = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_N \end{bmatrix} \quad (5.12)$$

Además para las pruebas bajo simulación se estableció que $\sigma_1 = \dots = \sigma_N = 2$ cm.

5.1.2.5. Diagrama de flujo

Con todas las consideraciones previas, el diagrama de flujo del método propuesto para la calibración de cada LPS independiente queda de la forma:

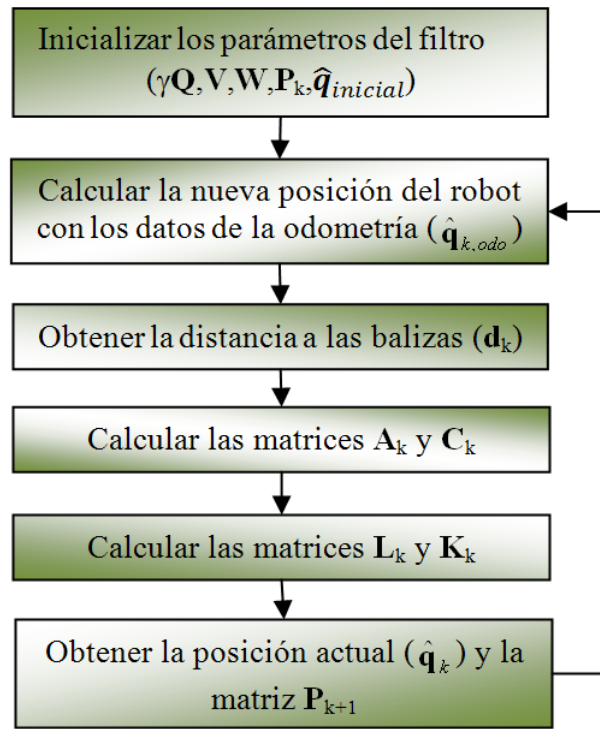


Figura 5.4: Diagrama de Flujo del método propuesto para la calibración de cada LPS.

5.1.3. Fase de Navegación

Esta fase comienza una vez finalizada la calibración de un LPS, en ella la posición de las balizas ya no son una incógnita del sistema y el filtro $H-\infty$ sólo tiene que estimar la posición y orientación del robot móvil. En esta situación las ecuaciones del filtro son las mismas, pero reduciéndose el número de componentes del vector de estado, ya que la posición de las balizas no forma parte de él. El nuevo vector de estado es:

$$\hat{\mathbf{q}}_{k-1} = [x_{k-1}, y_{k-1}, \theta_{k-1}]^T \quad (5.13)$$

La actualización de la posición del robot a través de los datos de odometría se obtiene ahora como:

$$\hat{\mathbf{q}}_{k,odo} = \begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = f(\hat{\mathbf{q}}_{k-1}, sistema) = \begin{bmatrix} x_{k-1} + \Delta D_k \cdot \cos(\theta_k) \\ y_{k-1} + \Delta D_k \cdot \sin(\theta_k) \\ \theta_{k-1} + \Delta \theta_k \end{bmatrix} \quad (5.14)$$

Quedando la nueva matriz de la dinámica de la odometría como:

$$\mathbf{A}_k = \frac{\partial \hat{\mathbf{q}}_{k,odo}}{\partial \mathbf{q}_k} \Big|_{\mathbf{q}_k = \hat{\mathbf{q}}_k} = \begin{bmatrix} 1 & 0 & -\Delta D_k \cdot \sin(\theta_k) \\ 0 & 1 & \Delta D_k \cdot \cos(\theta_k) \\ 0 & 0 & 1 \end{bmatrix} \quad (5.15)$$

La distancia desde el robot móvil a las balizas sigue siendo la misma que la expresada en la ecuación 5.9 pero al variar las componentes del vector de estado la matriz \mathbf{C}_k queda como:

$$\mathbf{C}_k = \frac{\partial \mathbf{d}_k}{\partial \mathbf{q}_k} \Big|_{\mathbf{q}_k = \hat{\mathbf{q}}_k} = \begin{bmatrix} \frac{\partial d_{1,k}}{\partial x_k} & \frac{\partial d_{1,k}}{\partial y_k} & \frac{\partial d_{1,k}}{\partial \theta_k} \\ \vdots & \vdots & \vdots \\ \frac{\partial d_{N,k}}{\partial x_k} & \frac{\partial d_{N,k}}{\partial y_k} & \frac{\partial d_{N,k}}{\partial \theta_k} \end{bmatrix} \quad (5.16)$$

Teniendo en cuenta estas consideraciones, el filtro diseñado para la etapa de calibración se puede emplear para la navegación del robot dentro de un LPS y no es necesario diseñar otro filtro para esta etapa.

5.2. Extensión a varios LPS

Una vez calibrado el primer LPS, el objetivo es conseguir calibrar otro sistema LPS dentro del edificio tal que comparta el origen de referencia con el primer LPS; de esta forma se consigue una calibración global de todos los sistemas (figura 5.5).

Para ello una vez finalizada la calibración del primer LPS se guía al robot hasta el segundo LPS y una vez que se llega a la nueva zona de cobertura se procede a realizar una nueva calibración.

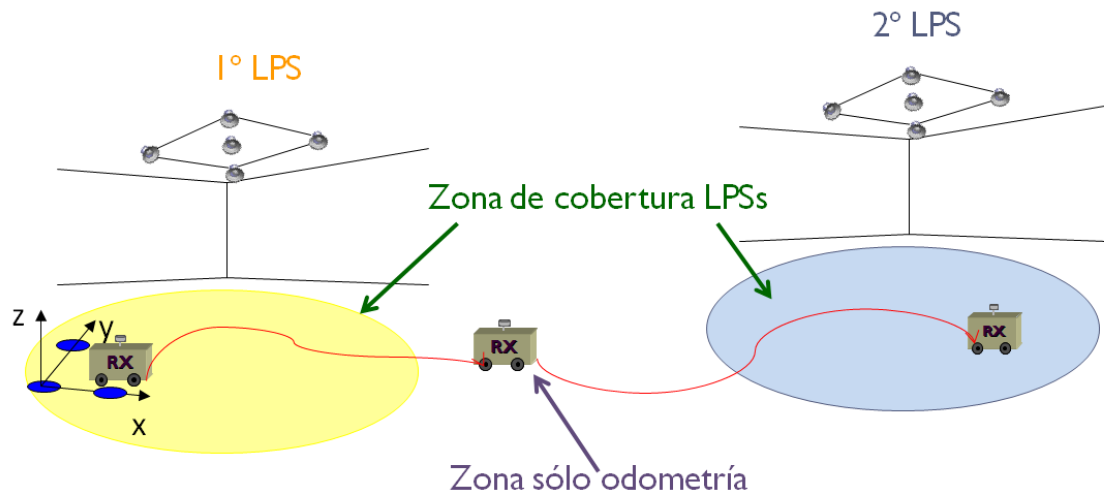


Figura 5.5: Calibración global de varios sistemas LPS

Durante el recorrido entre LPSs el robot móvil debe atravesar una región donde no hay apoyo de ningún LPS. En esa zona se sigue utilizando como sistema de navegación la información proveniente del filtro $H-\infty$ pero empleando sólo la fase de predicción, ya que al no haber información de distancia no se puede emplear la etapa de corrección. Como al llegar al nuevo LPS, la posición del robot no es muy fiable debido a que durante el trayecto no ha sido posible corregir los errores de odometría, se reinicializa el valor de la covarianza del filtro, \mathbf{P}_k , a un valor muy elevado.

Los errores producidos por la odometría en el recorrido entre los LPS, como se ha mencionado anteriormente, no se pueden corregir y afectan a la calibración del segundo LPS, ya que la desviación en ángulo y posición que existe entre la posición del punto real donde se inicia la calibración del segundo LPS y la posición del punto estimado, se traslada a la posición estimada de las balizas.

En la figura 5.6 se puede ver un ejemplo de este fenómeno. En esta figura la línea azul representa la trayectoria seguida por el robot, la línea verde la trayectoria estimada y los círculos las posiciones de las balizas: los azules las posiciones reales y los verdes las estimadas. Se han supuesto dos sistemas LPS y que en la calibración de los mismos no se comete ningún tipo de error ni en la odometría ni en la medida de los ultrasonidos, sólo existe error de odometría en la zona donde no hay cobertura del LPS. Se puede observar que si bien la estructura obtenida de la posición de las balizas del segundo LPS es correcta, aparece girada y desplazada de su posición real, este giro y desplazamiento coincide con el que existe entre el punto \mathbf{P}_2 real y el estimado. El punto \mathbf{P}_2 es el punto inicial de la calibración del segundo LPS.

Debido a que la estimación de este punto afecta de forma elevada al resultado de la calibración del segundo LPS, se proponen dos métodos para minimizar el error en la estimación del punto \mathbf{P}_2 .

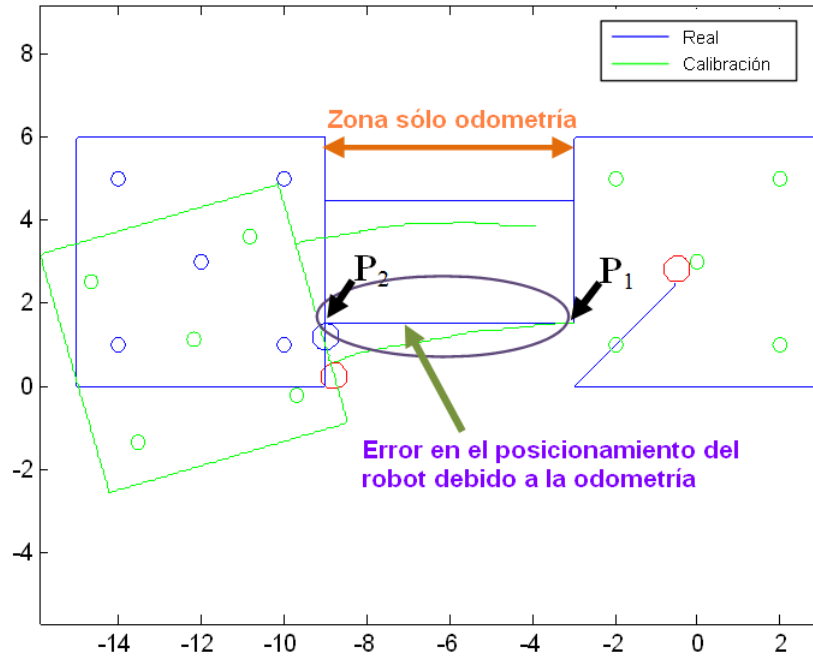


Figura 5.6: Error en la estimación de las balizas en el segundo LPS debido a los fallos cometidos en la odometría del robot. Los círculos son las proyecciones de las balizas en el plano de medida del robot..

5.2.1. Mejora de la estimación del punto P_2 con infraestructura adicional

Como la calibración de un sistema LPS es un proceso puntual y que no se requiere realizar habitualmente, se propone que durante esta fase se utilice un sistema LPS portable y de apoyo para minimizar las zonas donde no hay cobertura de ningún LPS (figura 5.7), reduciéndose así las zonas donde se acumulan los errores de odometría.

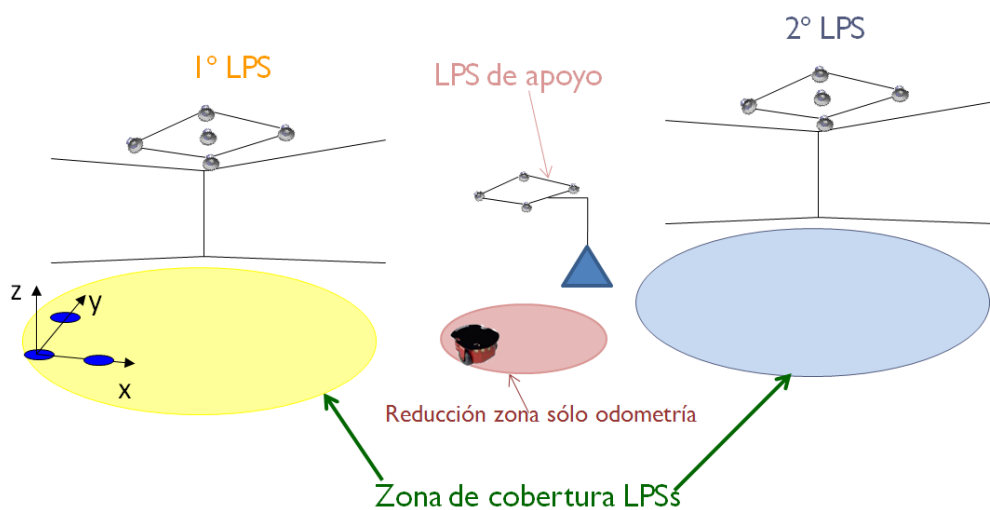


Figura 5.7: Empleo de un LPS de apoyo para reducir las zonas donde el robot navega sólo con odometría.

5.2.2. Mejora de la estimación del punto P_2 sin infraestructura adicional

Como emplear un elemento adicional, puede no ser una opción viable, en esta sección se describe un método que emplea el propio robot móvil para poder reducir el error en la estimación del punto P_2 . Para ello se realiza el siguiente proceso:

1. Una vez calibrado el primer LPS se lleva al robot a un punto P_1 , este será el punto de partida desde el primer LPS al segundo (ver figura 5.8). Cuando se llega a la zona de cobertura del nuevo LPS se detiene el robot y se almacena la posición del mismo (P_2) como punto de inicio de calibración del segundo LPS. Este punto no coincidirá con el real debido a los errores de odometría (Figura 5.8a).
2. A continuación se realiza el mismo recorrido que se realizó desde el primer LPS hasta el segundo pero marcha atrás. De esta forma el robot vuelve al primer LPS y se puede emplear la información almacenada durante este recorrido para crear un recorrido virtual desde el primer LPS al segundo, obteniendo así una segunda estimación del punto P_2 (Figura 5.8b).
3. Como la posición del robot no es la correcta debido a los errores de odometría, a través de la información del primer LPS, se obtiene la ubicación real del robot, ya que este LPS está bien calibrado y se lleva al punto P_1 . Desde ese punto se vuelve a repetir el proceso tantas veces como requiera el diseñador (Figura 5.8c).
4. Para calcular la posición final del punto P_2 (figura 5.8d) se calcula en primer lugar la media y la desviación típica de todos los puntos almacenados en el proceso, se eliminan los puntos almacenados que disten de la posición media obtenida en alguna de sus componentes (x, y, z) más de dos veces la desviación típica calculada, de esta forma se eliminan los peores puntos. Con los puntos restantes se calcula de nuevo la posición media y ese punto es la estimación definitiva de P_2 (Figura 5.8d).

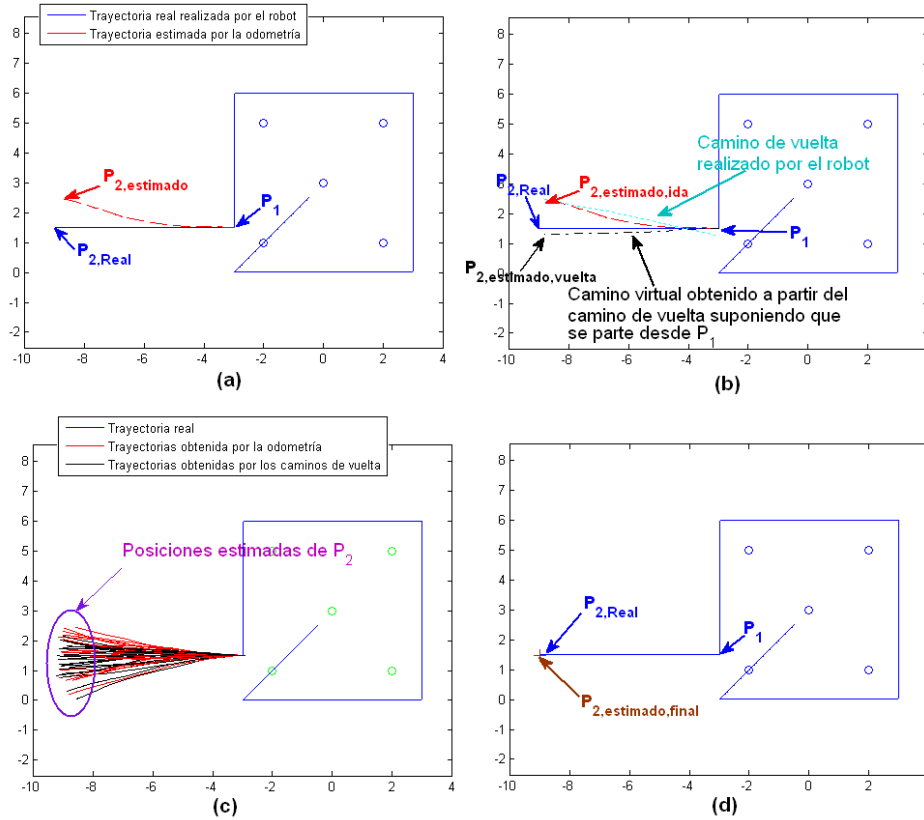


Figura 5.8: Mejora en la estimación del punto P_2 realizando varias veces el viaje de ida y vuelta entre los LPS con el robot móvil.

5.2.3. Diagrama de flujo

Teniendo en cuenta todo lo presentado anteriormente en la figura 5.9 se muestra el diagrama de flujo completo del sistema propuesto:

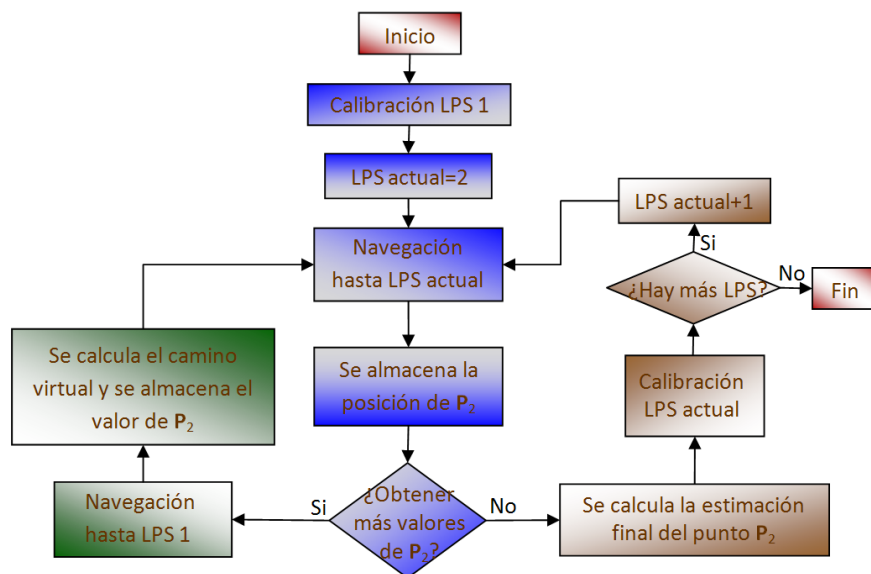


Figura 5.9: Diagrama de flujo para varios LPS.

5.3. Pruebas realizadas bajo simulación

En esta sección se presentan varias pruebas realizadas en un entorno de simulación en las que se analizan las propiedades del algoritmo propuesto en el capítulo. En un primer apartado se va a comprobar el funcionamiento cuando el sistema ha calibrado las balizas y sólo realiza funciones de navegación, a continuación se muestran diferentes pruebas de calibración, primero para un único LPS y luego para dos LPS.

5.3.1. Guiado del robot móvil

En estas pruebas se supondrá que se conoce la posición de las balizas y únicamente se guía al robot dentro del LPS combinando la información de la odometría y de las distancias al LPS. En las pruebas propuestas se compara el error medio cometido entre el recorrido real, el recorrido obtenido con el filtro H- ∞ , el recorrido realizado por un filtro de Kalman extendido y el cometido por la odometría.

Para obtener el error medio, se almacena para cada uno de los métodos tanto el recorrido realizado como el ángulo del robot en cada uno de los puntos. Se calcula el error en cada componente que existe entre la posición y el ángulo obtenido con el robot en cada uno de los puntos con su posición y ángulo real, a continuación el error medio en posición y ángulo se obtiene con la media de todos esos errores.

Si se expresan las matrices de posiciones y ángulos del robot tanto reales, (\mathbf{X}_r) como estimadas (\mathbf{X}_m) en la realización de un recorrido de N puntos como:

$$\begin{aligned}\mathbf{X}_r &= \begin{bmatrix} x_{r,1} & x_{r,2} & \cdots & x_{r,N} \\ y_{r,1} & y_{r,2} & \cdots & y_{r,N} \\ \theta_{r,1} & \theta_{r,2} & \cdots & \theta_{r,N} \end{bmatrix} \\ \mathbf{X}_m &= \begin{bmatrix} x_{m,1} & x_{m,2} & \cdots & x_{m,N} \\ y_{m,1} & y_{m,2} & \cdots & y_{m,N} \\ \theta_{m,1} & \theta_{m,2} & \cdots & \theta_{m,N} \end{bmatrix}\end{aligned}\quad (5.17)$$

entonces el error en cada una de las componentes puede expresarse como:

$$\mathbf{e} = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{bmatrix} = \mathbf{X}_r - \mathbf{X}_m \quad (5.18)$$

y el error medio entre el método empleado y las posiciones reales como:

$$\mathbf{e}_m = \begin{bmatrix} e_{m,x} \\ e_{m,y} \\ e_{m,\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{N} \sum_{n=1}^N e_{1,n} \\ \frac{1}{N} \sum_{n=1}^N e_{2,n} \\ \frac{1}{N} \sum_{n=1}^N e_{3,n} \end{bmatrix} \quad (5.19)$$

Para estas pruebas la posición de las balizas es la fijada en la tabla 5.1

Tabla 5.1: Posición de las balizas para las pruebas de guiado del robot.

Posición (metros)	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0	-2	2	-2	2
Coordenada Y	3	1	1	5	5
Coordenada Z	3	3	3	3	3

Además se fijan las siguientes condiciones para la simulación:

- El período de muestreo (T_s) es: 100 ms.
- La desviación típica de ruido en las medidas del LPS es de: 3 cm.
- La desviación típica en la velocidad lineal del robot es de: 1mm (cada T_s), esto es, 1 cm/s.
- La desviación típica del ángulo del robot es de: 0.0573° (cada T_s), esto es, 0.573%/s.
- El valor de γ es: $\gamma = 10^{-4}$

Para la primera prueba se va a realizar un recorrido circular dentro del área de cobertura

- Posición inicial real del robot móvil: [0, 0,0]
- Posición inicial del robot indicada al algoritmo: [-2.0, 0.5, 0];

En la figura 5.10 se muestran los resultados obtenidos con los diferentes métodos durante el recorrido, en azul la trayectoria real, en rojo la trayectoria obtenida por el filtro de Kalman extendido, en verde la trayectoria calculada con el filtro H_∞ y por último en cian la trayectoria calculada por la odometría. De forma visual se puede observar cómo los métodos que fusionan tanto la información de odometría como la información proporcionada por el LPS consiguen aproximarse al recorrido real del robot.

En la figura 5.11 se muestra un zoom de la parte superior del recorrido, se puede apreciar más claramente como el filtro H_∞ es el método que más se aproxima al trazado real.

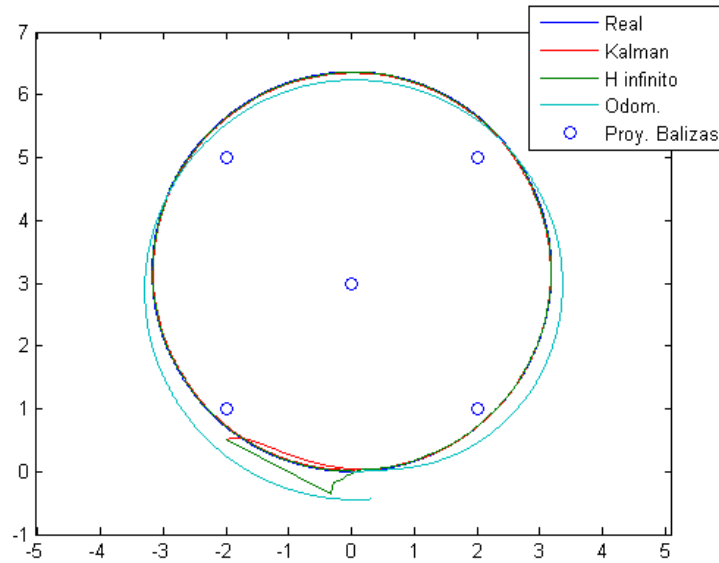


Figura 5.10: Recorrido circular realizado en el primer test de navegación.

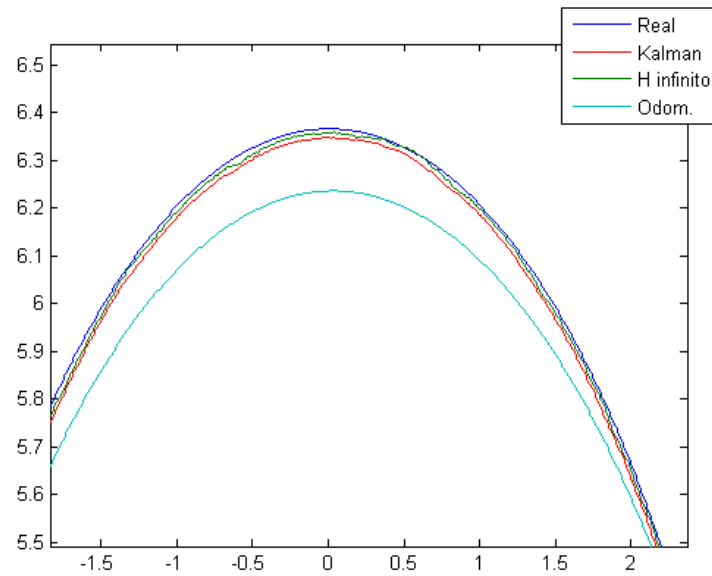


Figura 5.11: Ampliación de la parte superior del recorrido realizado en el primer test de navegación.

En la Tabla 5.2 se muestran los errores medios obtenidos por los diferentes métodos. Se puede comprobar que la odometría es el método que más error comete ya que no aprovecha la información proporcionada por el LPS. También se comprueba que el método que más se aproxima al trazado real es el filtro $H-\infty$, que comete un error medio en posición de 1.65 cm y un error de ángulo de 0.06 radianes. También se ha comprobado que el método basado en el filtro $H-\infty$ converge más rápido que el método basado en el filtro de Kalman extendido.

Tabla 5.2: Errores medios obtenidos con los diferentes métodos en el primer test de guiado.

	error medio 'X'	error medio 'Y'	error medio ' θ '
Odometría	15.74 cm.	26.12 cm.	0.50 rd.
Kalman extendido	7.28 cm.	5.29 cm.	0.07 rd.
Filtro H- ∞	1.60 cm.	1.64 cm.	0.06 rd.

En la segunda prueba se realiza un recorrido cuadrado dentro del área de cobertura, para esta prueba las condiciones de simulación son las mismas que en el caso anterior excepto:

- Posición inicial real del robot móvil: [-3, 0,0]
- Posición inicial del robot indicada al algoritmo: [-3.2, 0.5, 0];

En la figura 5.12 se muestran los resultados obtenidos en el nuevo recorrido por los diferentes métodos, se puede observar cómo, al igual que en el caso anterior, los métodos que fusionan la información de la odometría y del LPS son los que más se acercan al trazado real. Además en la figura 5.13 se muestra un zoom de una de las esquinas y se vuelve a comprobar que el método basado en el filtro H- ∞ se acerca más a la posición real que el método basado en el filtro de Kalman extendido. Además en esta prueba se puede observar como el filtro H- ∞ converge mucho más rápido a la posición real que el método de Kalman

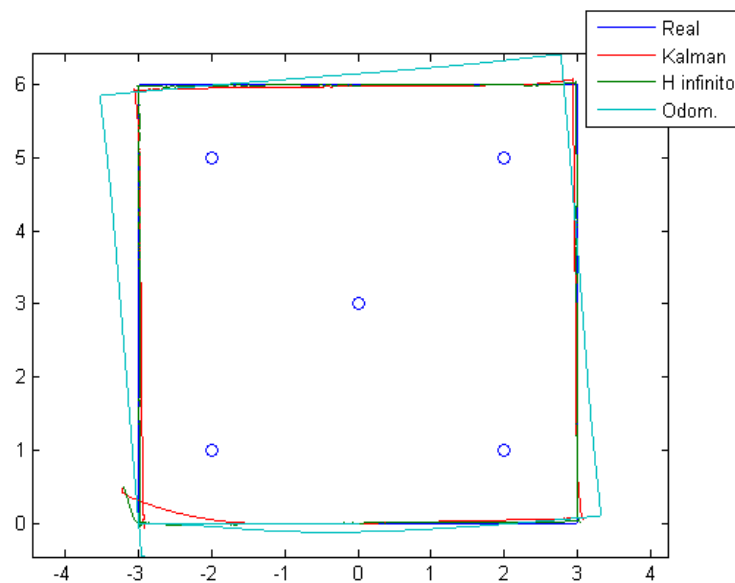


Figura 5.12: Recorrido cuadrado realizado en el segundo test de navegación

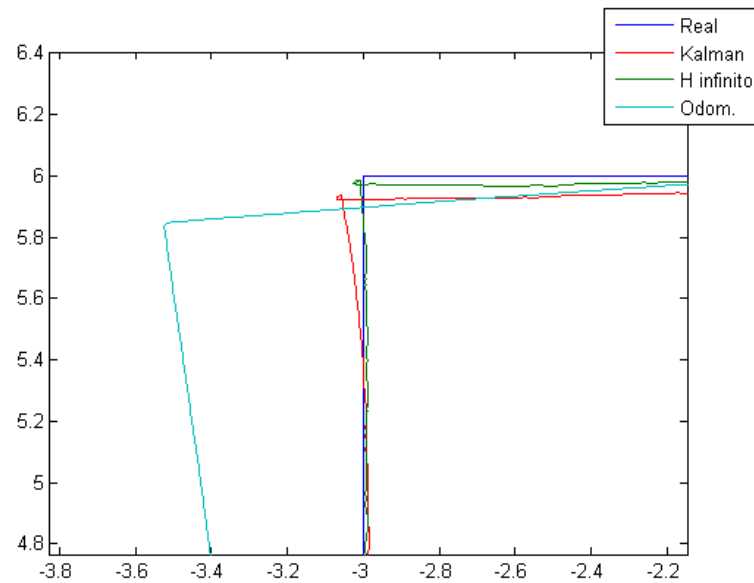


Figura 5.13: Ampliación de la esquina superior izquierda del recorrido realizado en el segundo test de navegación.

Por último en la tabla 5.3 se muestran los errores medios obtenidos con cada uno de los métodos; se vuelve a comprobar que los métodos que fusionan la información de odometría obtienen un mejor resultado y que el filtro H- ∞ con errores medios inferiores a 1.7 cm se acerca más a la posición real del robot que el filtro de Kalman extendido con errores en torno a los 5 cm.

Tabla 5.3: Errores medios obtenidos con los diferentes métodos en el segundo test de guiado.

	error medio 'X'	error medio 'Y'	error medio ' θ '
Odometría	22,54 cm.	20,03 cm.	0.64 rd.
Kalman extendido	5.15 cm.	5.35 cm.	0.08 rd.
Filtro H- ∞	1.69 cm.	1.52 cm.	0.06 rd.

5.3.2. Calibración de un sistema LPS individual

En esta sección se muestran varias pruebas realizadas bajo simulación donde se realiza la calibración de un sólo sistema LPS.

En la primera prueba, las posiciones de las balizas son las indicadas en la tabla 5.1 y las condiciones de simulación son las mismas que en los apartados anteriores.

En la figura 5.14 se muestra el recorrido realizado en la zona de cobertura. Para poder visualizar mejor los resultados se muestra una representación en dos dimensiones de los mismos; aunque la calibración de las balizas es en tres dimensiones, en la figura se muestra la proyección de las balizas en el plano de medida del robot. En la figura, las líneas azules representan el recorrido real, las líneas verdes el recorrido estimado y los círculos rojos las balizas después de la fase de Precalibración.

Se puede apreciar de forma visual cómo el recorrido estimado se aproxima al real y como la fase de Calibración mejora la posición de las balizas obtenida en la fase de Precalibración.



Figura 5.14: Recorrido realizado en el área de cobertura en el primer test de calibración. Los círculos son las proyecciones de las balizas en el plano de medida.

Para evaluar de forma empírica la mejora introducida por la fase de calibración se va a emplear el error medio cometido en la estimación de las balizas. Para ello, si se tiene un sistema formado por N balizas y se define $br_i = (br_{x,i}, br_{y,i}, br_{z,i})$ $i = 1 \dots N$ como la posición real de la baliza i y $be_{ik} = (be_{x,ik}, be_{y,ik}, be_{z,ik})$ $i = 1 \dots N$ la posición de la baliza estimada por el algoritmo en la iteración k de la baliza i , entonces el error medio cometido en la iteración k se calcula como:

$$em_k = \frac{1}{N} \sum_{i=1}^N \sqrt{(br_{x,i} - be_{x,ik})^2 + (br_{y,i} - be_{y,ik})^2 + (br_{z,i} - be_{z,ik})^2} \quad (5.20)$$

En la figura 5.15 se muestra el error medio cometido en la estimación de la posición de las balizas a lo largo del proceso de calibración: el primer valor corresponde al error en las balizas después de la fase de Precalibración (6.27 cm), y el último es el valor final del error después de la fase de Calibración (2.81 cm). Se puede observar cómo mejora la estimación de las balizas ya que se reduce a más de la mitad el error cometido en la estimación. Se da por finalizada la etapa de calibración cuando la variación de la posición de las balizas estimadas no sea superior a 10^{-4} en las últimas 50 iteraciones. En la tabla 5.4 se muestra la posición final de las balizas estimadas.

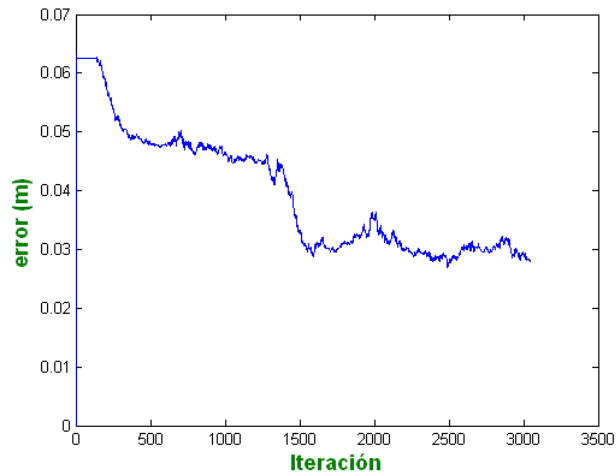


Figura 5.15: Error medio cometido en la posición de las balizas estimadas en el primer test de calibración.

Tabla 5.4: Posición final de las balizas estimadas en el primer test de calibración.

Posición (metros)	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.129	-1.994	2.010	-1.987	2.014
Coordenada Y	2.993	0.983	0.988	4.990	4.999
Coordenada Z	3.012	2.997	3.005	3.002	3.004

Además, como se ha visto en la figura 5.14, el algoritmo no sólo realiza la calibración sino que también obtiene la posición del robot durante todo el recorrido, por eso en la figura 5.16 se muestra el error cometido en la posición (a) y orientación (b) del robot en cada uno de los instantes de muestreo de la fase de Calibración. Se puede observar que la posición obtenida del robot se aproxima a la real ya que el error máximo cometido es inferior a los 8 centímetros en posición y de 0.04 radianes en ángulo durante todo el proceso y además el error medio cometido en cada componente $[x(cm), y(cm), \theta(rd)]$ durante la prueba es de: $[2.81\text{ cm}, 2.47\text{ cm}, 0.017\text{ rd}]$.

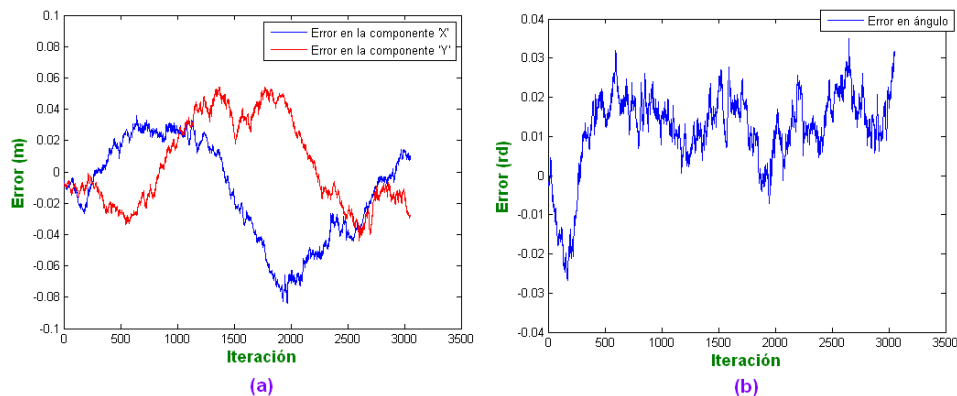


Figura 5.16: Error cometido en posición y ángulo del robot durante la fase de calibración en el primer test.

A continuación se repite la misma prueba 100 veces y en la figura 5.17 se muestran los valores de desviación típica en las tres componentes X, Y, Z obtenidas en cada una de las balizas. Se puede observar que la fase de Calibración reduce la desviación típica en el error haciendo que en la mayoría de los casos esté por debajo de los 3 cm. También cabe destacar que en la componente Z es donde se produce una reducción del error más significativa, ya que las desviaciones típicas después de realizar la Calibración están por debajo de 1 cm.

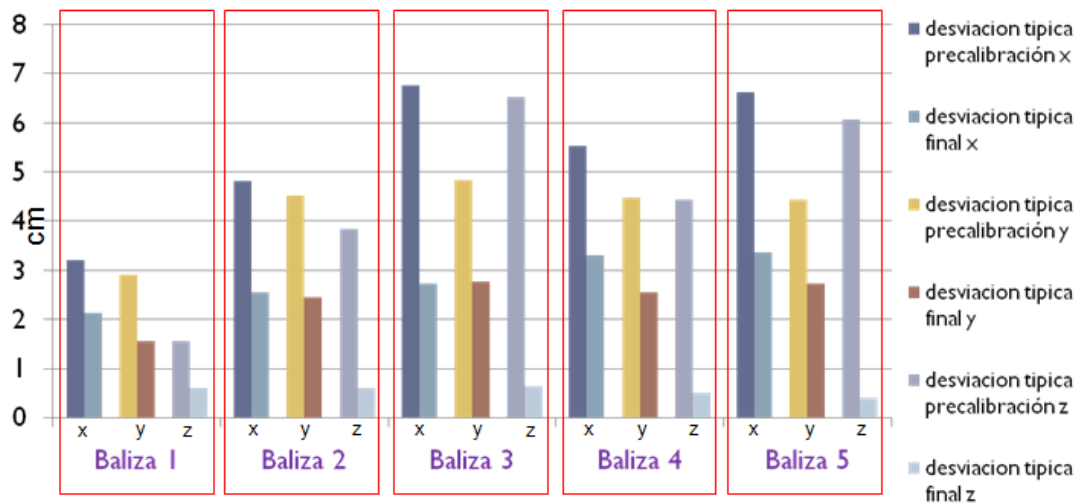


Figura 5.17: Desviación típica en el error en cada una de las componentes X, Y, Z de cada una de las balizas obtenidas después de realizar 100 simulaciones con el recorrido de la primera calibración.

Por último en la figura 5.18 se comparan las elipses de error con un intervalo de confianza del 95% obtenidas después de la fase de Precalibración y después de la fase de Calibración. Se muestran sólo las elipses para las coordenadas X e Y para una mejor visualización y además, como se ha visto en la figura 5.17, los errores en la componente Z son bastante menores. Se puede observar que una vez finalizada la etapa de calibración las elipses se reducen consiguiendo una mejor estimación de la posición de las balizas.

A continuación se realiza el mismo test pero en este caso se van a introducir dos deslizamientos en la trayectoria: en el primer deslizamiento se simula el derrape de una rueda durante 0.5 segundos en el recorrido lo que provoca un error de ángulo en la odometría (I); el segundo deslizamiento consiste en un derrape de las dos ruedas del robot durante 2 segundos, simulando de esa forma un error en la posición de la odometría (II).

En la figura 5.19 se muestra el recorrido realizado durante la prueba y la posición de las balizas obtenidas tanto en la fase de Precalibración como en la fase de Calibración, además se marcan las zonas donde se producen los derrapes. Al igual que en el caso anterior, aunque se realiza una calibración en 3 dimensiones, para poder visualizar mejor los resultados se muestran en la figura únicamente las componentes X e Y de las balizas. Se puede observar cómo el algoritmo es capaz de seguir la posición del robot durante el

recorrido a pesar de los derrapes y que la posición de las balizas estimadas se aproxima más a la real que la obtenida en la fase de Precalibración. En azul se representa el recorrido real, en rojo el obtenido por la odometría y en verde el estimado.

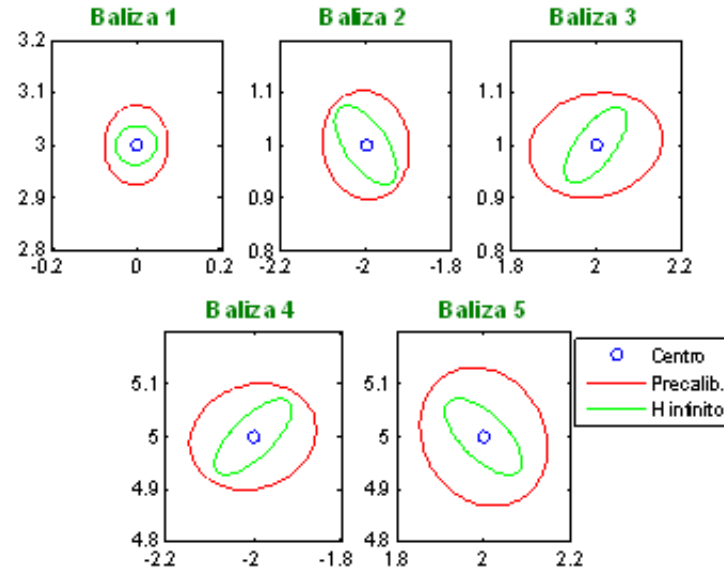


Figura 5.18: Cálculo de las elipses de error con un intervalo de confianza del 95 % en la posición estimada de las balizas en el primer test. En rojo las elipses obtenidas después de la fase de Precalibración, en verde las obtenidas después de la fase de Calibración.

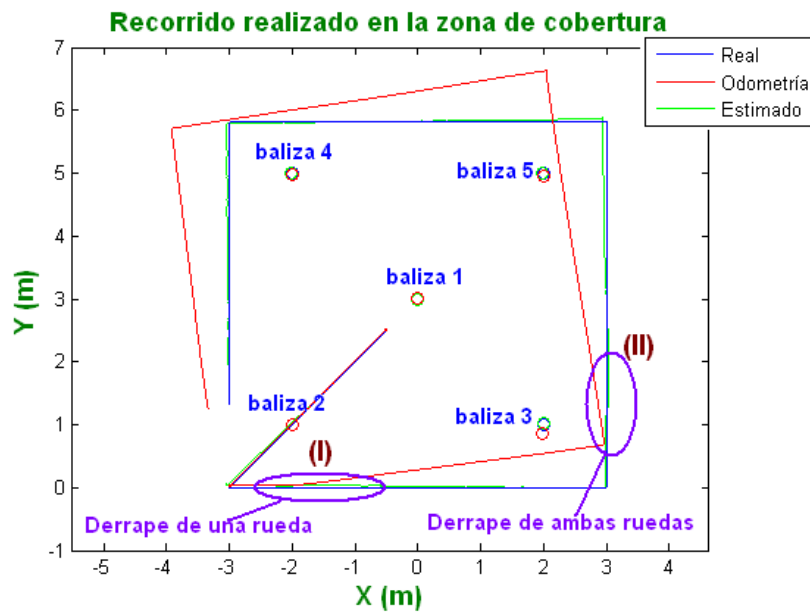


Figura 5.19: Recorrido realizado en el área de cobertura en el segundo test de calibración. Los círculos son las proyecciones de las balizas en el plano de medida.

El error medio cometido en la posición de las balizas en cada punto del recorrido se muestra en la figura 5.20. Se puede observar que, a pesar de los derrapes introducidos en el recorrido, el error cometido al final del proceso, 2.38 cm, es menor que el error en la localización de las balizas después de la fase de Precalibración, 6.02 cm. Además se puede apreciar el aumento en el error en el posicionamiento de las balizas cuando se producen los derrapes y cómo el sistema es capaz de recuperarse de estos fallos. En la tabla 5.5 se muestra la posición final de las balizas estimadas.

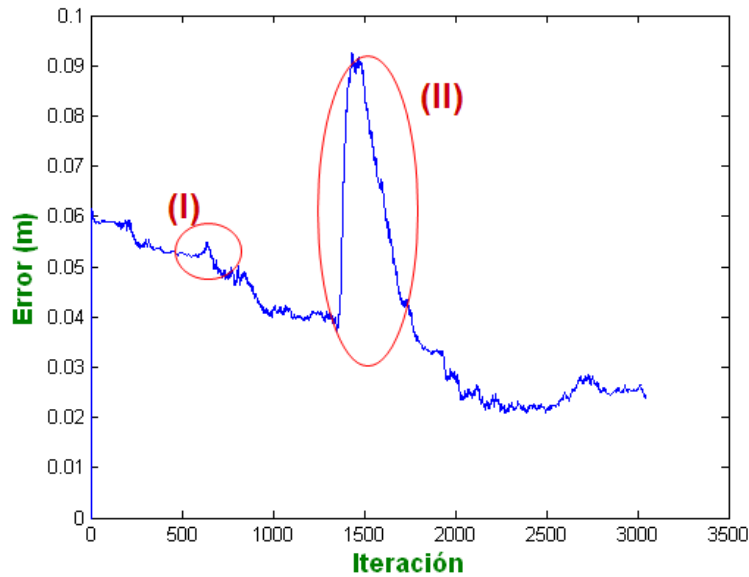


Figura 5.20: Error medio cometido en la posición de las balizas estimadas en el segundo test de calibración.

Tabla 5.5: Posición final de las balizas estimadas en el segundo test de calibración.

Posición (metros)	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.000	-1.986	2.010	-2.010	1.974
Coordenada Y	2.996	0.993	1.020	4.996	5.010
Coordenada Z	2.978	2.993	2.972	2.994	2.988

Como en el caso anterior en la figura 5.21 se muestra el error en la posición (a) y ángulo (b) del robot en cada uno de los instantes del recorrido; en el caso de la posición se puede observar que el error debido al derrape en las dos ruedas afecta mucho más que el error en el ángulo ya que justo en ese momento el error aumenta hasta 17 cm. En el caso del ángulo, el error que más le afecta es el angular ya que en ese instante el error aumenta hasta los 0.11 rd. Además el sistema demuestra que es capaz de recuperarse de los errores y volverse a acercar a la posición real del robot. En este caso los errores medios obtenidos en cada componente $[x(cm), y(cm), \theta(rd)]$ durante la prueba aumentan debido a los errores impulsivos de la odometría: $[3.05\text{ cm}, 3.55\text{ cm}, 0.019\text{ rd}]$.

Se realiza la misma prueba 100 veces y en la figura 5.22 se muestran las desviaciones típicas obtenidas en cada una de las componentes de la posición de las balizas una vez realizadas las 100 simulaciones. Se observa que a pesar de los derrapes en la odometría la fase de Calibración sigue mejorando los resultados obtenidos en la fase de Precalibración. Se sigue observando que la reducción más significativa se produce en la componente Z

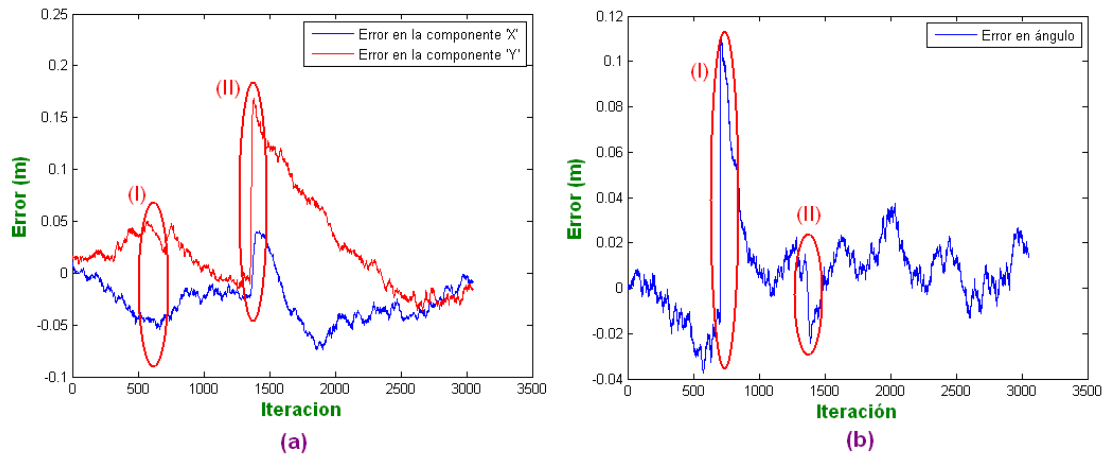


Figura 5.21: Error cometido en posición y ángulo del robot durante la fase de calibración en el segundo test.

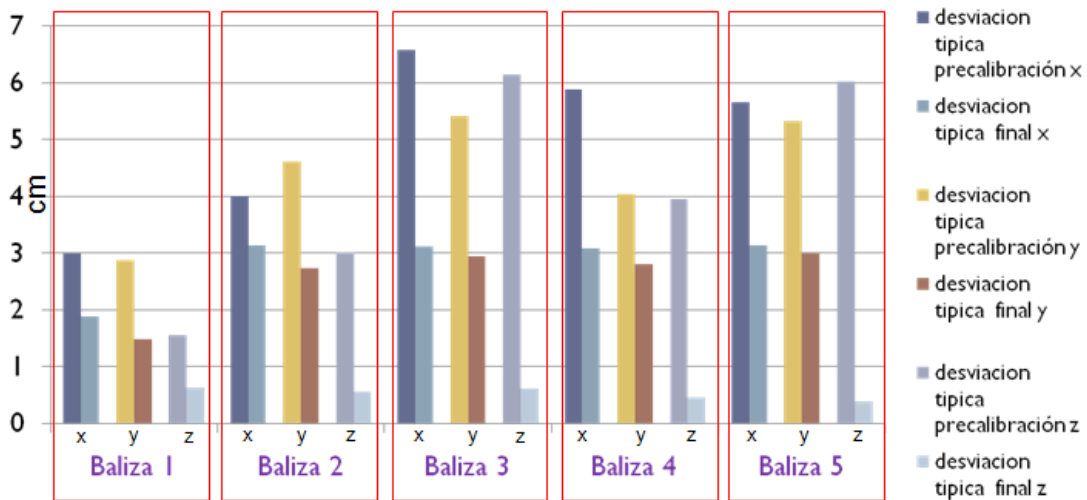


Figura 5.22: Desviación típica en el error en cada una de las componentes X,Y,Z de cada una de las balizas obtenidas después de realizar 100 simulaciones con el recorrido de la segunda calibración.

Por último en la figura 5.23 se muestran las elipses de error con un intervalo de confianza del 95 %. Al igual que en el caso anterior, debido a que el error en la componente Z es menor, sólo se muestran los resultados para las componentes X e Y. Se puede observar cómo, a pesar de los deslizamientos en la odometría, las elipses son más pequeñas después de la fase de Calibración.

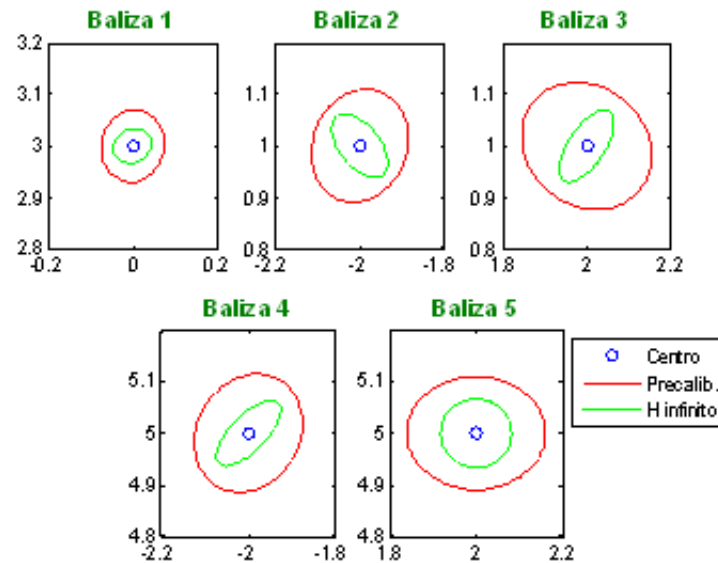


Figura 5.23: Cálculo de las elipses de error con un intervalo de confianza del 95 % en la posición estimada de las balizas en el segundo test. En rojo las elipses obtenidas después de la fase de Precalibración, en verde las obtenidas después de la fase de Calibración.

5.3.3. Calibración global de varios sistemas LPS

En esta sección se muestran resultados bajo simulación en la calibración de dos sistemas LPS independientes. Para ello, al sistema de posicionamiento empleado en las anteriores pruebas se le añade un segundo sistema cuyas balizas estan situadas según se muestra en la tabla 5.6.

Tabla 5.6: Posición real de las balizas del segundo LPS.

Posición (metros)	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	-12	-14	-10	-14	-10
Coordenada Y	3	1	1	5	5
Coordenada Z	3	3	3	3	3

Dentro de los LPS las condiciones de simulación son las mismas que en los apartados anteriores, pero para magnificar los problemas producidos en la zona donde el robot sólo navega con la odometría, esto es, donde no hay cobertura de ningún sistema LPS, se van a suponer los siguientes errores:

- La desviación típica de error en la velocidad lineal del robot cuando se desplaza sólo con odometría es: 4 cm/s.
- La desviación típica de error en la velocidad angular del robot cuando se desplaza sólo con odometría es: 2.292°/s.

Además la etapa de refinamiento se efectua 10 veces.

En la figura 5.24 se muestra el recorrido realizado en la calibración del primer LPS. Al igual que en la primera prueba para LPS individuales, aunque se realiza una calibración tridimensional de las balizas, para poder visualizar mejor los resultados sólo se hace una representación en dos dimensiones.

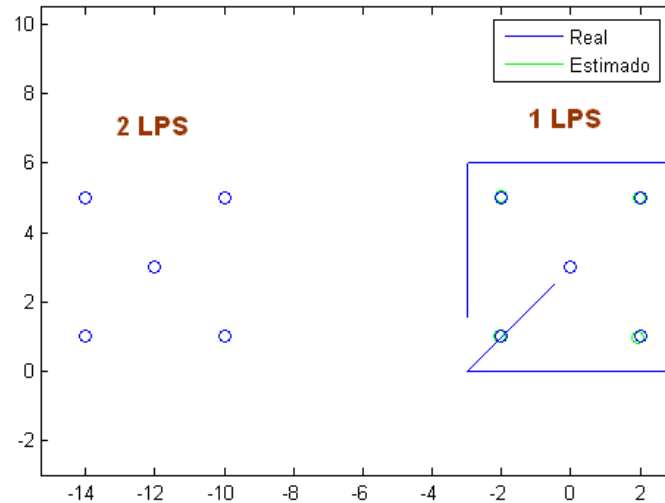


Figura 5.24: Recorrido realizado en la calibración del primer LPS en el primer test de calibración global. Los círculos son las proyecciones de las balizas en el plano de movimiento del robot.

El error medio cometido en la posición de las balizas en la primera calibración es de 2.81 cm comprobándose que los resultados son similares a los del apartado anterior. Además los errores por componentes en cada una de las balizas se muestran en la tabla 5.7

Tabla 5.7: Error en la posición final de las balizas estimadas el el primer LPS para el primer test de calibración global.

error (metros)	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.019	0.033	0.045	0.004	0.004
Coordenada Y	0.009	0.032	0.009	0.034	0.003
Coordenada Z	0.001	0.005	0.001	0.005	0.005

A continuación se procede al proceso de estimación del punto de inicio de calibración del segundo LPS. Esto se muestra en la figura 5.25 donde se aprecian los diferentes caminos recorridos (a) y la estimación final del punto de calibración (b).

En el cálculo de dicho punto de inicio se comete un error final en cada componente $[x(m), y(m), \theta(rd)]$ de $[0.004m, 0.033m, 0.008rd]$ respectivamente. Además en la tabla 5.8 se muestran tanto las coordenadas reales como estimadas de dicho punto.

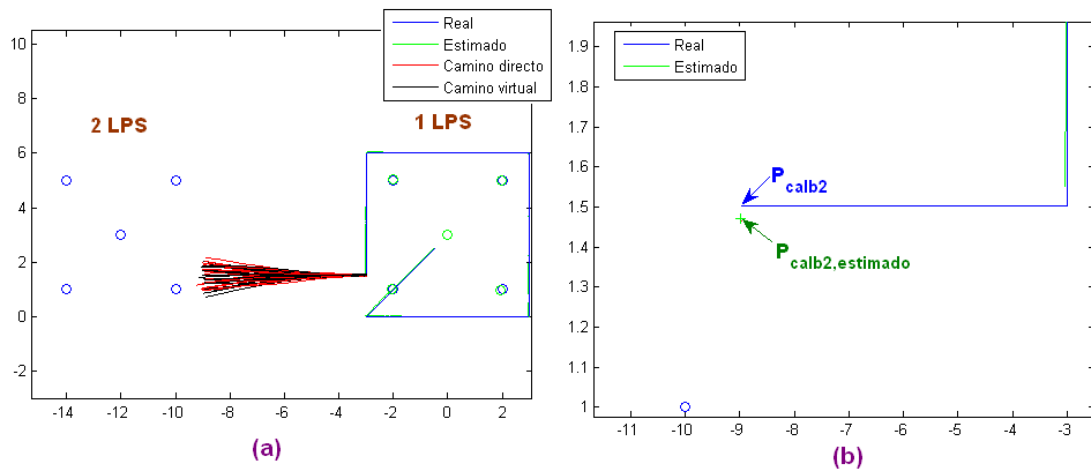


Figura 5.25: Estimación del punto inicial de calibración del 2º LPS para la primera prueba de calibración global. Los círculos son las proyecciones de las balizas en el plano de movimiento del robot.

Tabla 5.8: Posición tanto real como estimada del punto inicial de calibración del segundo LPS en la primera prueba de calibración global.

Posición	Coordenada X (m)	Coordenada Y (m)	Ángulo (θ) (rd)
Posición real	-8.996	1.504	-3.1416
Posición estimada	-8.992	1.470	-3.1332

Una vez obtenido el punto de inicio se procede a la calibración del segundo LPS. En la figura 5.26 se muestra el resultado de dicha calibración; además también se muestra, en línea negra, el resultado que se hubiera obtenido si no se hubiese empleado el método de cálculo del punto inicial de calibración. Se puede observar visualmente que el proceso de estimación mejora los resultados de calibración.

En la tabla 5.9 se muestran en centímetros los errores cometidos en cada una de las componentes de cada baliza, tanto si se utiliza el método de mejora como si no se emplea. Se puede observar la mejora en los resultados que se obtiene si se emplea el método de estimación del punto inicial de calibración del segundo LPS. También se muestra que el proceso de mejora sólo modifica las componentes X e Y ya que la componente Z no varía.

Tabla 5.9: Error en cm de la estimación de la posición de las balizas del segundo LPS en la primera prueba de calibración global.

Error (cm)	Con Mejora			Sin Mejora		
	x	y	z	x	y	z
Baliza 1	5.04	7.56	0.21	33.05	78.32	0.21
Baliza 2	7.97	8.71	0.25	15.39	99.66	0.25
Baliza 3	6.02	4.48	0.5	11.56	60.83	0.5
Baliza 4	2.76	9.55	0.2	53.36	97.05	0.2
Baliza 5	3.14	5.68	0.38	51.82	57.60	0.38

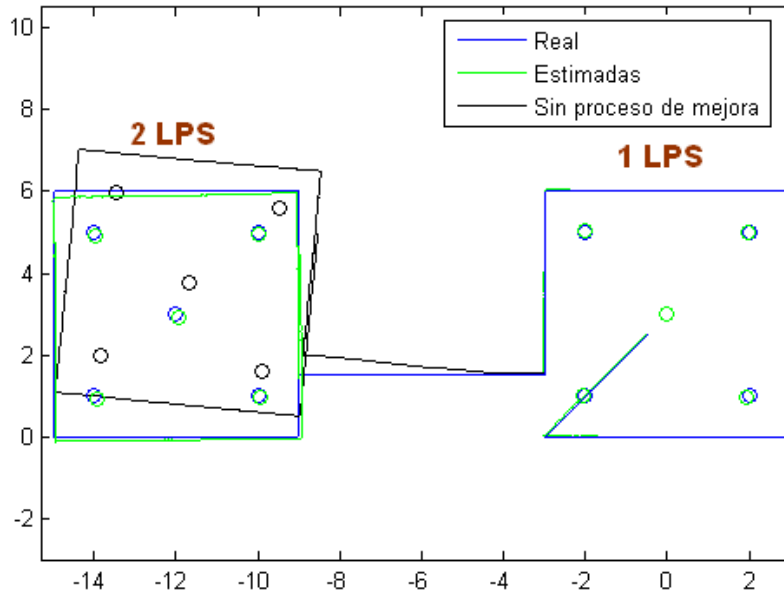


Figura 5.26: Calibración global del 2º LPS tanto si se emplea el método de estimación del punto inicial de calibración como sino se emplea. Los círculos son las proyecciones de las balizas en el plano de medida del robot.

En el siguiente test de calibración global se va a suponer que las balizas de cada LPS independiente se encuentran situadas según las posiciones indicadas en la tabla 5.10.

Tabla 5.10: Localización real de las balizas en ambos LPS en la segunda prueba de calibración global.

Posición (m)	1º LPS			2 LPS		
	x	y	z	x	y	z
Baliza 1	1.8	1.5	3.0	-7.4	3.8	3.0
Baliza 2	-0.5	4.0	3.0	-6.7	3.5	3.0
Baliza 3	-0.1	2.2	3.0	-9.2	2.5	3.0
Baliza 4	-1.2	2.0	3.0	-8.1	2.7	3.0
Baliza 5	-1.9	2.85	3.0	-7.5	2.2	3.0

Las condiciones de simulación del robot y de los LPS son las mismas que en la primera prueba, sólo que en este caso se varía el recorrido realizado dentro del área de cobertura de cada LPS. En la figura 5.27 se muestra el proceso de calibración completo; al igual que en el caso anterior se muestra cómo la mejora de la estimación del punto inicial de calibración del segundo LPS hace que los resultados sean también mejores.

Para esta simulación los errores obtenidos en las posiciones de las balizas de ambos LPS, tanto si se emplea el método de mejora de la estimación del punto inicial de calibración como sino no se utiliza, se muestran en la tabla 5.11. Como se puede observar los errores cometidos siempre que se realice el proceso de mejora están por debajo de los 5 cm, mientras que si no se efectúa este proceso los errores cometidos en las componentes X e Y suben al orden de decímetros.

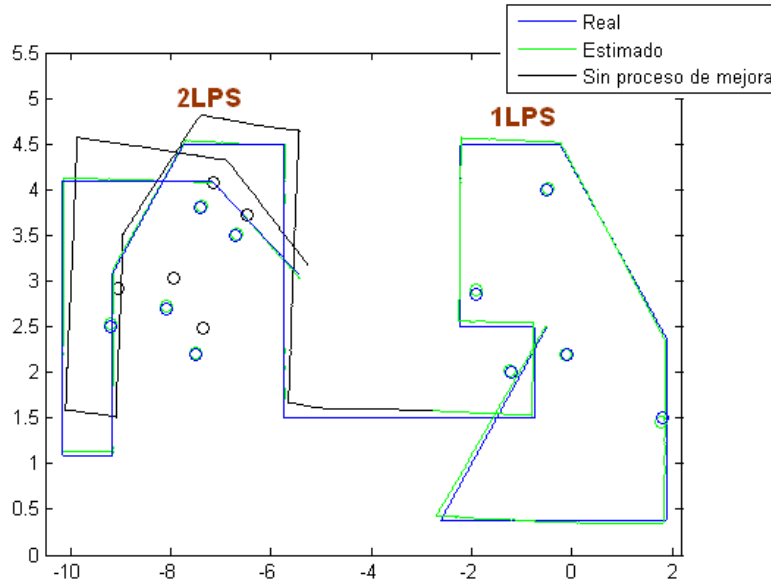


Figura 5.27: Calibración global del sistema para el segundo test tanto si se emplea el método de estimación del punto inicial de calibración como si no se emplea. Los círculos son las proyecciones de las balizas en el plano de medida del robot.

Tabla 5.11: Error en cm de la estimación de la posición de las balizas en ambos LPS

Error (cm)	1º LPS			2 LPS con mejora			2 LPS sin mejora		
	x	y	z	x	y	z	x	y	z
Baliza 1	3.26	4.36	0.51	2.19	2.13	0.36	26.31	28.20	0.36
Baliza 2	1.26	0.76	1.06	1.86	0.66	0.13	23.70	22.11	0.13
Baliza 3	2.48	0.48	0.16	0.58	3.66	0.34	15.30	42.33	0.34
Baliza 4	3.72	2.41	0.29	0.26	2.37	0.20	17.15	33.53	0.20
Baliza 5	1.23	4.64	0.67	0.26	1.89	0.35	13.61	29.13	0.35

5.4. Conclusiones

En este capítulo se ha presentado un método de calibración global para varios LPS independientes mediante un robot móvil. En una primera fase el algoritmo realiza la calibración del primero de los LPS mediante un filtro $H-\infty$; a continuación, una vez finalizada la calibración, el robot móvil avanza hasta el siguiente LPS y realiza la calibración de este nuevo sistema respecto al sistema de referencia del primer LPS.

La fase de calibración de cada LPS independiente se divide en tres fases:

- Fase de Precalibración. Donde a través de un recorrido del robot se realiza un posicionamiento inverso de las balizas.
- Fase de Calibración. Mediante un filtro $H-\infty$ se fusionan los datos de odometría y la información del LPS para refinar la posición de las balizas calculadas en la fase de Precalibración.

- Fase de Navegación. Una vez calibradas las balizas se adapta el filtro $H-\infty$ para navegar con el robot dentro del área de cobertura.

Además, cuando el robot navega de un LPS calibrado a otro que no lo está, puede navegar por zonas donde no hay cobertura de ningún LPS. En estas zonas los errores introducidos por la odometría influyen gravemente en la calibración de los nuevos LPS. Por eso se ha propuesto un mecanismo para reducir dichos errores; de los resultados obtenidos en las simulaciones se pueden extraer varias conclusiones:

- Se ha comprobado como la fase de calibración mejora notablemente los errores cometidos en la posición de las balizas por la fase de precalibración, consiguiendo que los errores originales se reduzcan casi a la mitad en las coordenadas X e Y y a menos de 1 cm en la componente Z . Además se ha comprobado que el algoritmo es robusto a deslizamientos en las medidas de odometría, ya que si bien en los momentos de los deslizamiento el error aumenta, si se sigue moviendo el robot dentro de la zona de cobertura el sistema se recupera de dicho error.
- Se ha comprobado que el método propuesto para reducir la influencia de los errores de la odometría en las zonas donde el robot no tiene apoyo de ningún LPS consigue reducir el error de calibración de forma significativa, ya que los errores originales con un orden de magnitud en torno a decímetros se reducen a errores en torno a centímetros. En estas zonas debido a la influencia del error de odometría se recomienda mover al robot lo más recto y despacio posible para evitar deslizamientos.
- Una vez realizada la calibración del sistema se ha comprobado el funcionamiento del filtro $H-\infty$ como método de guiado del robot dentro del sistema. Se ha comprobado su correcto funcionamiento y sus mejores propiedades frente a otro sistema empleado típicamente en el guiado de robots móviles como es el filtro de Kalman extendido.

Pruebas Reales

En esta sección se presentan los resultados obtenidos en pruebas reales para cada uno de los métodos propuestos en los capítulos anteriores: en el primer apartado se describe el sistema de posicionamiento local empleado para la realización de las pruebas; a continuación en el segundo apartado se muestran los resultados obtenidos para las pruebas de posicionamiento basándose en los bideterminantes de Cayley-Menger; en la tercera sección se presentan los resultados para las pruebas de autocalibración basadas en medidas desde diferentes puntos de test; seguidamente en el punto cuarto se muestran las pruebas para el algoritmo de autocalibración global basado en el filtro $H-\infty$; en el último punto se analizan las conclusiones extraídas de este capítulo.

6.1. Estructura del sistema LPS real

La arquitectura básica del LPS donde se realizan las pruebas experimentales en espacios interiores se inspira en el funcionamiento del GPS utilizado en exteriores. Un número no limitado de objetos móviles son capaces de calcular su posición mediante técnicas de trilateración a través de las señales recibidas desde las balizas. Para ello parten de los tiempos de vuelo medidos tras las emisiones efectuadas por una serie de balizas situadas en posiciones conocidas del entorno. Sin embargo a diferencia del sistema GPS que sólo emplea multiteración esférica, el LPS desarrollado permite emplear tanto trilateración esférica (empleando una señal de sincronismo cuando emiten las balizas), o trilateración hiperbólica (donde no se emplea dicha señal).

En la figura 6.1 puede verse un esquema de dicho LPS. Lo componen 5 balizas situadas en el techo y sincronizadas vía hardware mediante un cable que las une. A cada baliza se le asocia un código binario único que ha sido escogido de modo que tenga una baja correlación cruzada con los que identifican a las otras balizas. De este modo se consigue que las cinco balizas puedan emitir simultáneamente sin interferirse unas con otras, y que los receptores sean capaces de distinguir la procedencia de las emisiones.

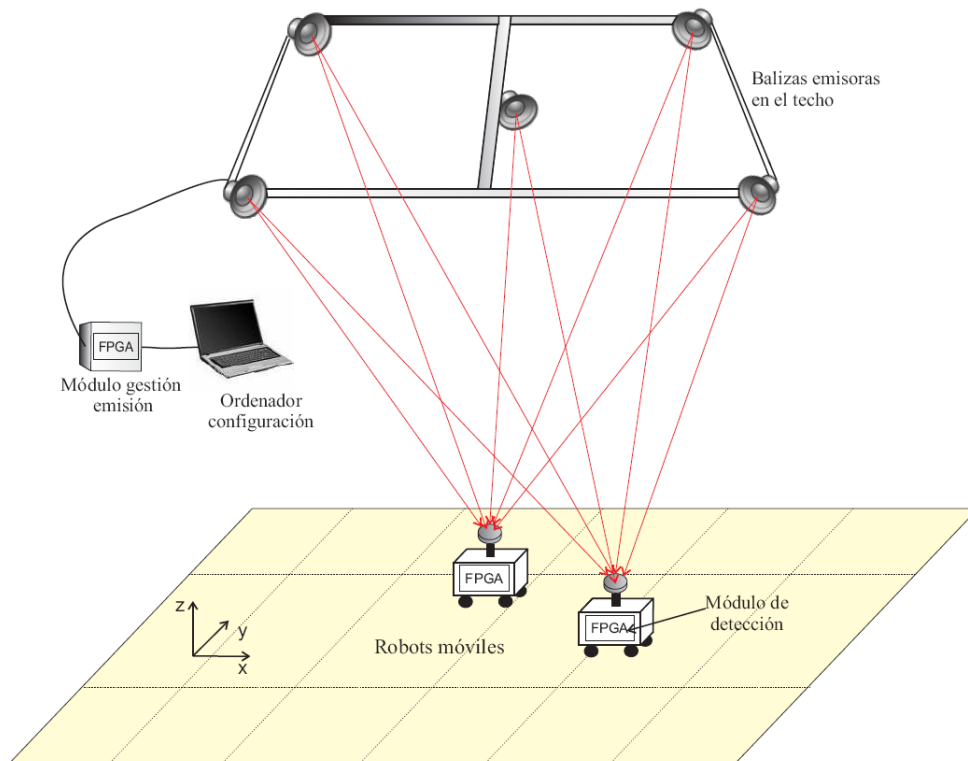


Figura 6.1: Esquema del sistema LPS

En la figura 6.2 puede encontrarse una representación funcional de los elementos de la figura 6.1. Se resume a continuación la utilidad de cada uno de los componentes implicados.

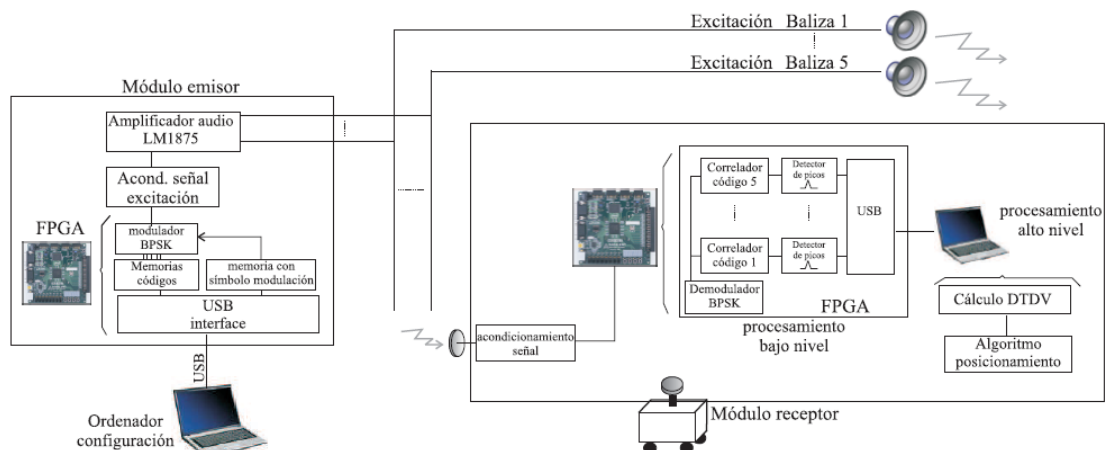


Figura 6.2: Esquema funcional del sistema LPS

- **Cinco balizas situadas en posiciones fijas en el techo.** Las balizas se encuentran a 3.50 metros del suelo y están dispuestas sobre una estructura metálica de $1\text{ m} \times 1\text{ m}$ que sirve de soporte. Este elemento constituye básicamente toda la infraestructura a instalar para el funcionamiento del sistema. Además al situar las balizas en el techo no se entorpece el área de posicionamiento y su cableado no supone un problema.

En la figura 6.3 se muestra la infraestructura con las balizas (B1 a B5) y la separación que existe entre los transductores. Como se demostró en [Rub09] la disposición de los transductores es tal que si se emplean códigos ortogonales generalizados, esto es códigos LS, con una ventana libre de interferencias (IFW) de tamaño superior a 173 bits ($W \geq 86$), la diferencia entre los instantes de llegada de las emisiones procedentes de cada baliza en cualquier punto del área de trabajo no supera el tamaño de dicha IFW (asumiendo una velocidad de propagación de los ultrasonidos en aire de $c = 345.5 \text{ m/s}$ y un único ciclo de portadora de frecuencia en torno a los 40 kHz). Se consigue además con esta disposición maximizar la zona de cobertura común entre balizas.

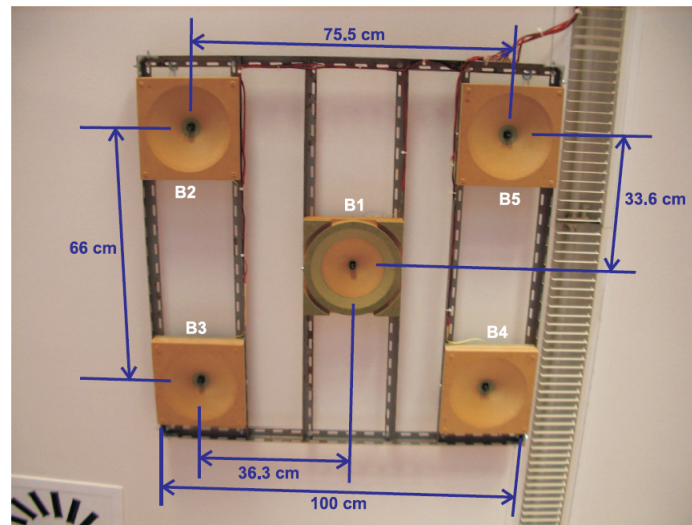


Figura 6.3: Estructura de las balizas

Respecto a los transductores, se han utilizado emisores cilíndricos de PVDF de MSI [Inc08a] con frecuencia de resonancia de 40 kHz y un ancho de banda de 8 kHz . Cada transductor se ha insertado en una estructura cónica que actúa como reflectora y permite aumentar el área de cobertura sobre el suelo [VUM⁺07, PUH⁺07].

■ **Ordenador de configuración.** Para dotar al sistema de la mayor versatilidad posible se ha incluido un ordenador que permite configurar varios parámetros de emisión:

- Los códigos a emitir por las balizas y la longitud de los mismos.
- La forma de onda y número de ciclos de la portadora que constituyen el símbolo de modulación BPSK.
- Frecuencia de la señal de portadora.
- Período de repetición de las emisiones.
- Habilitación/Deshabilitación de la emisión.

Este ordenador se conecta vía USB con la FPGA del bloque emisor para la configuración de los parámetros anteriores.

- **Modulo emisor.** Este bloque es el encargado de atacar a las balizas emisoras y se compone de dos submódulos. El primero se ha implementado sobre una FPGA Spartan3E de Xilinx [Xil08] y permite generar los códigos a emitir. Consiste en una serie de memorias RAM y un modulador BPSK. En la fase de inicialización, el ordenador de configuración graba en las memorias los códigos a emitir y la señal portadora, especificando además las frecuencias de emisión y el periodo de repetición de los códigos. Se activa después la señal de activación del sistema y se inicia la modulación BPSK de los códigos almacenados a la frecuencia especificada en la configuración. Las señales resultantes son adaptadas en un segundo módulo para la excitación del transductor con los niveles que indica el fabricante. En la figura 6.4 se puede observar la estructura del módulo emisor incluyendo el ordenador de configuración.

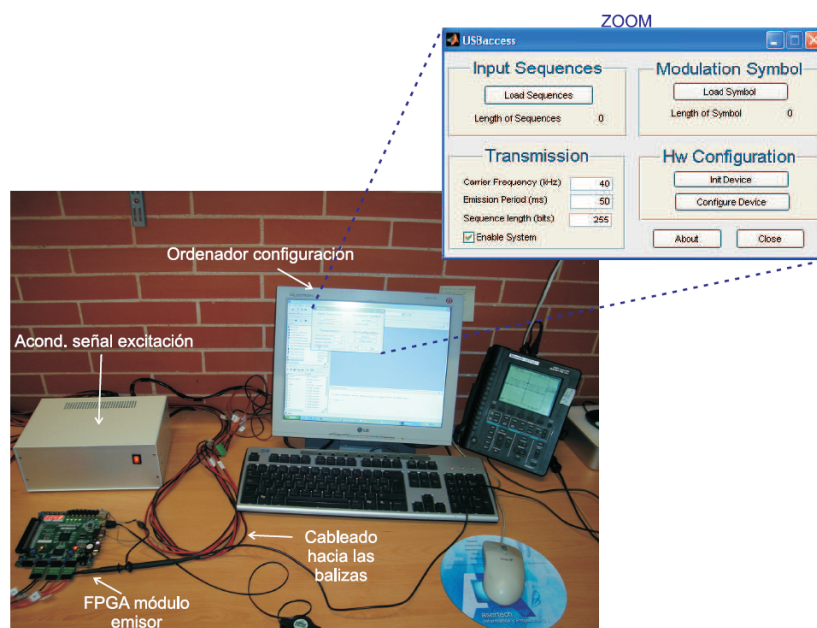


Figura 6.4: Modulo emisor.

- **Receptores móviles.** La arquitectura de todos los receptores es la misma y consta de los siguientes módulos:
 - Captura de las señales emitidas y ajuste de escala para mantener constante los niveles de señal que entrega el micrófono utilizado (Panasonic Electret 61B [Cor08]).
 - Procesamiento de bajo nivel. Las señales capturadas son demoduladas en BPSK de modo asíncrono. A continuación un bloque de correlación localiza en la señal recibida los códigos emitidos por cada una de las balizas. El último bloque

detecta el valor de los picos máximos en cada correlación y el instante de tiempo en el que se ha producido. Todo este procesado se realiza en una FPGA Spartan3E de Xilinx.

- Procesamiento de alto nivel. La información relativa a los picos de correlación es enviada a través del USB al ordenador a bordo del robot en donde se calculan los tiempos de vuelo y se realiza el posicionamiento. Existe además la opción de enviar los datos de salida del bloque detector de picos via WiFi al ordenador de configuración. Aunque esta opción puede considerarse que va en contra de la privacidad de este tipo de sistemas, en donde el módulo receptor no emite ninguna señal para que sólo él conozca su posición, se ha permitido durante la fase de pruebas a efectos de depuración, ya que el sistema a bordo del robot no dispone de pantalla para visualizar los resultados.

El robot utilizado en las pruebas es el PIONEER 3-DX8 [Inc08b] utilizado en multitud de aplicaciones docentes y de investigación, al que se le ha añadido una pequeña plataforma para la sujeción del micrófono, su electrónica asociada y la sujeción de la FPGA. En la figura 6.5 se puede ver a uno de estos robots dentro de la zona de trabajo.

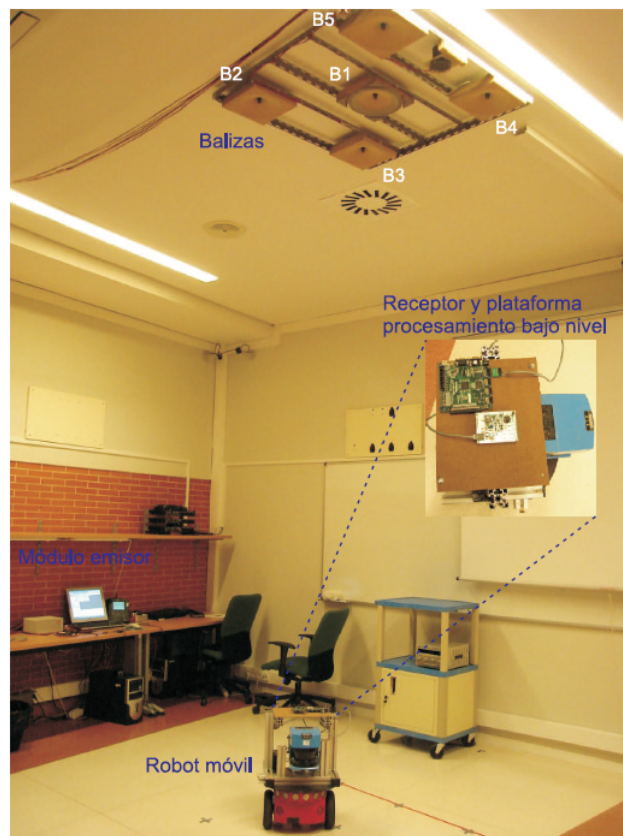


Figura 6.5: Escenario de las pruebas realizadas con el LPS ultrasónico.

- **Área de trabajo.** Las pruebas han sido realizadas en una habitación de 3.45 m de altura organizado según muestra la figura 6.6. El área de cobertura se ha restringido a un rectángulo sobre el suelo de $4.50\text{ m} \times 3.50\text{ m}$ justo debajo de las balizas y dividido en cuadrados de 50 cm de lado.

En las siguientes secciones se describen con mayor profundidad cada uno de los bloques funcionales que componen el LPS.

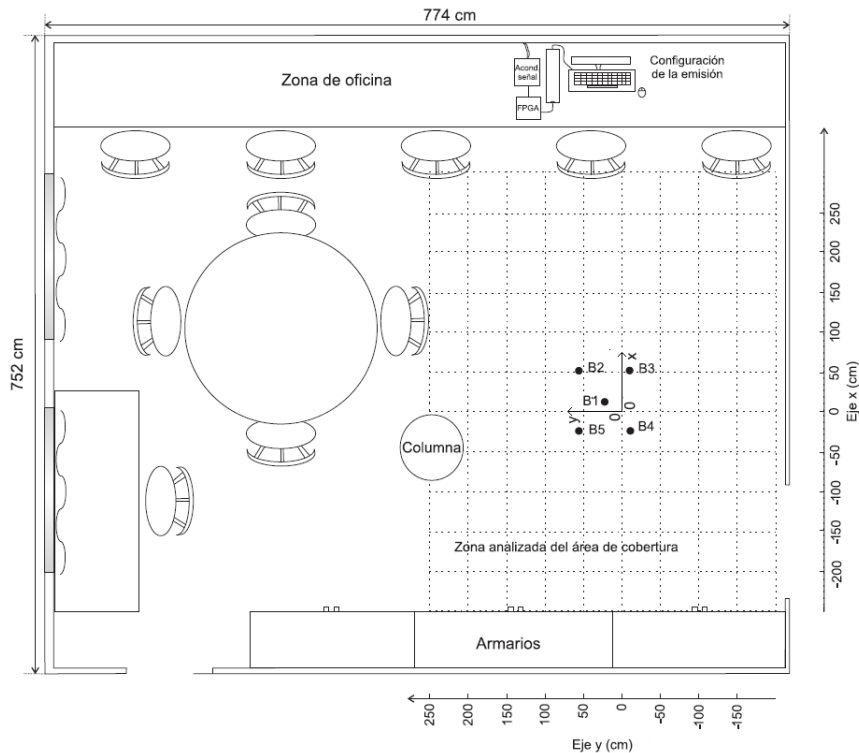


Figura 6.6: Esquema de la zona de trabajo. Además se muestra la proyección de las balizas en el suelo y el origen de coordenadas.

6.1.1. Características de los transductores ultrasónicos empleados

6.1.1.1. Transductor en la etapa de emisión

Los emisores usados en las balizas son de MSI [Inc08a] y sus propiedades piezoeléctricas se deben a una lámina de PVDF de $30\text{ }\mu\text{m}$ de espesor encajada sobre un soporte cilíndrico. En la figura 6.7 se puede ver el aspecto y principales características de estos transductores. Su patrón de emisión es omnidireccional en el plano horizontal, mientras que en el vertical la directividad es de $\pm 40^\circ$ (con una pérdida de -6 dB respecto a los 0°). La frecuencia de resonancia se encuentra en torno a los 40 kHz con un ancho de banda de 8 kHz . (figura 6.7)

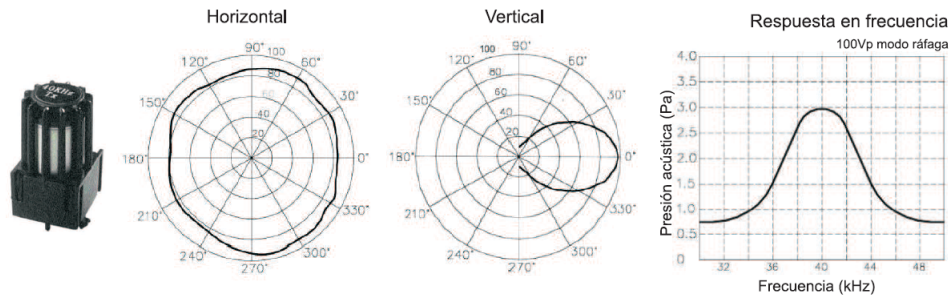


Figura 6.7: Características del emisor utilizado: patrones de emisión horizontal, vertical y respuesta en frecuencia.

Teniendo en cuenta el ángulo de apertura en el plano axial del transductor, así como la disposición en el techo de las balizas y en el suelo de los receptores, resulta necesario redirigir las emisiones de cada transductor hacia el suelo cubriendo la mayor zona posible. Para ello, se ha utilizado la estructura propuesta en [VUM⁺07] que consiste en un reflector cónico de madera en donde se inserta el transductor. Los parámetros que definen el reflector (ángulo del cono, radio y distancia del vértice al transductor) se han diseñado tratando de evitar que haces correspondientes a una misma emisión pero reflejados en distintas zonas del cono se reciban superpuestos en el receptor. En el LPS descrito anteriormente, para una baliza situada a 3.45 m de altura y un ángulo de incidencia en el suelo del haz de 40°, se puede garantizar un área circular de cobertura de radio 3.20 m con una atenuación máxima de 10dB.

6.1.1.2. Transductor en la etapa de recepción

En la etapa de recepción se emplea un micrófono omnidireccional *electrect* Panasonic WM-61B [Cor08] diseñado para aplicaciones de audio de proposito general. Alguna de sus principales características son su reducido tamaño, su omnidireccionalidad y el amplio rango de frecuencias que es capaz de detectar (figura 6.8). Se ha probado en el laboratorio que su respuesta es prácticamente plana entre los 20 Hz y 45 kHz.

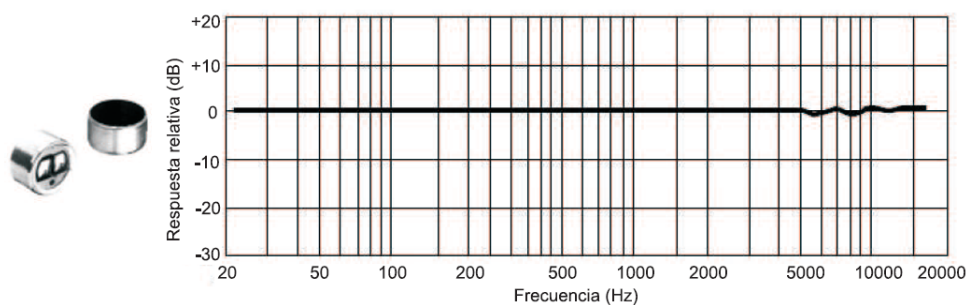


Figura 6.8: Respuesta en frecuencia del micrófono empleado en la recepción.

Para realizar la adaptación de la señal adquirida por el micrófono a los niveles de señal requeridos por el sistema de digitalización se ha empleado el circuito integrado SSM2166 [Dev08]. Este integrado consta de una etapa de preamplificación seguida de un control automático de ganancia, cuya función consiste en mantener constantes los niveles de señal que entrega el micrófono.

6.1.2. Módulo emisor

Como ya se ha mencionado en los apartados anteriores, el módulo emisor tiene como misión generar las señales codificadas de cada baliza y adaptarlas según las especificaciones de los transductores empleados. En la figura 6.9 se muestra el diagrama de bloques de este módulo. El sistema está configurado para trabajar por defecto con macro-secuencias de 1024 bits generadas mediante concatenación, en donde cada bit del código se modula con un símbolo compuesto por un único ciclo de una portadora cuadrada de frecuencia $f_e = 41.667\text{ kHz}$, siendo el periodo de repetición de las emisiones $T_R = 50\text{ ms}$. Sin embargo, todos estos parámetros pueden configurarse en tiempo de ejecución a través del ordenador de configuración. El ordenador comunica estos datos vía USB a la plataforma de desarrollo: una Nexys2 de Digilent.

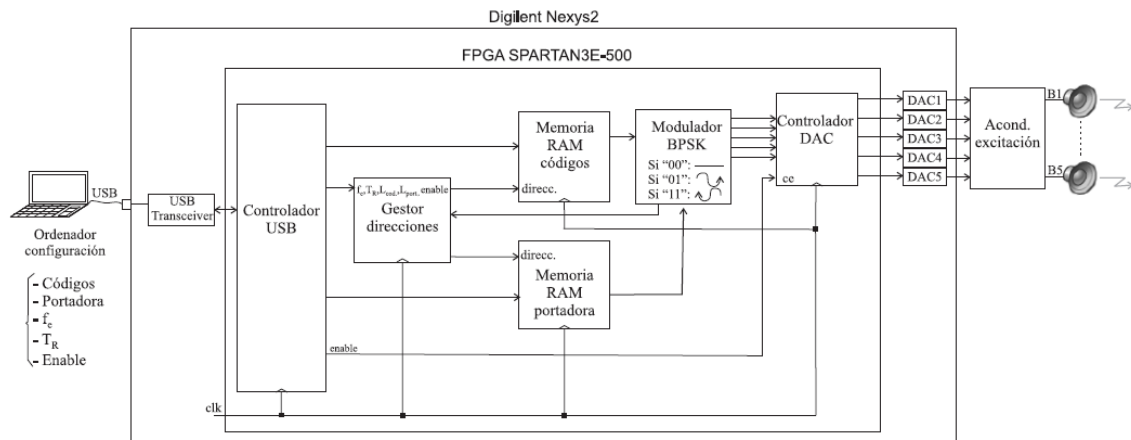


Figura 6.9: Diagrama de bloques del módulo emisor.

6.2. Pruebas de posicionamiento con el algoritmo propuesto basado en Cayley-Menger

Para comprobar las propiedades de los algoritmos propuestos, se realizan varias pruebas y se comparan los resultados con el algoritmo de posicionamiento de resolución de las ecuaciones de distancia mediante el algoritmo de Gauss-Newton (GN, ver anexo A) ya que es uno de los que mejores resultados ofrece [YTB08]. Se realizan dos grupos de pruebas: en el primero se toman 50 medidas en varios puntos fijos dentro del área de cobertura y se comparan los resultados con los métodos propuestos LZC y LTI, mientras que en el segundo se realiza una ruta dentro del área de cobertura.

El primer paso para poder realizar las pruebas consiste en medir la posición de las balizas respecto al eje de referencia establecido en la figura 6.6, para ello se procedió a realizar una calibración manual del sistema (este proceso se explicará en la sección 6.3) encontrándose que las balizas se encontraban en las posiciones indicadas por la tabla 6.1.

Tabla 6.1: Posición de las balizas en el espacio de trabajo.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.122 m.	0.513 m.	0.515 m.	-0.24 m.	-0.242 m.
Coordenada Y	0.224 m.	0.56 m.	-0.1 m.	-0.11 m.	0.56 m.
Coordenada Z	3.5 m.	3.5 m.	3.5 m.	3.5 m.	3.5 m.

Como se ha descrito en las secciones anteriores el LPS disponible tiene las balizas a la misma altura, esto es, en un plano Z constante, en este caso a 3.5 m. Por lo tanto las pruebas se realizan para los algoritmos y métodos siguientes (descritos en el capítulo 3) :

1. Pruebas para el algoritmo de posicionamiento esférico basado en Cayley-Menger y comparativa con el método GN.
2. Pruebas para el método LZR (método 1) de posicionamiento hiperbólico propuesto en la sección 3.3.1 y comparativa con el método GN.
3. Pruebas para el método LTI (método 3) propuesto en la sección 3.3.3 y comparativa con el método GN.

Los algoritmos ejecutan un posicionamiento 3D, aunque sólo se visualizarán resultados sobre el plano donde se han efectuado las medidas ya que, dada la configuración de las balizas, los errores cometidos en estimación de la coordenada z son mayores; se ha elegido esta configuración porque el objetivo final es un guiado en las coordenadas X-Y sobre el suelo.

6.2.1. Pruebas en puntos fijos

A) Posicionamiento esférico

Para la realización de estas pruebas se tomarán 50 medidas en varios puntos de test dentro del área de cobertura. Estos puntos están separados aproximadamente medio metro unos de otros y tratan de abarcar todo el área de cobertura disponible.

En la figura 6.10 se pueden observar los resultados obtenidos empleando el algoritmo de trilateración esférica. Se puede ver cómo la diferencia entre el método esférico basado en los bideterminantes de Cayley-Menger y el método GN esférico es inferior a 1.5 mm excepto en tres de los puntos de medida; en estos tres puntos hay mayores diferencias debido a que son regiones donde se producen recepciones con multitrayecto y las medidas de tiempos de vuelo contienen un error más elevado.

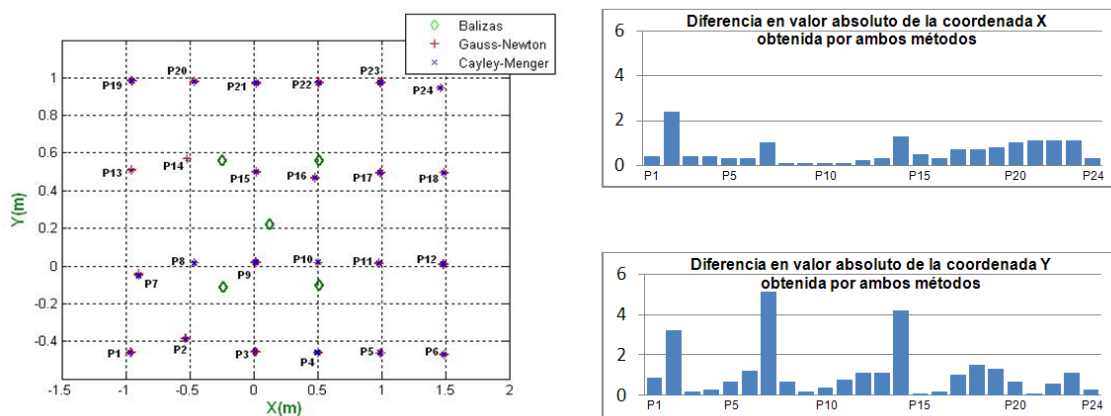


Figura 6.10: Comparativa entre el método esférico basado en los bideterminantes de Cayley-Menger y el método GN esférico para una rejilla de puntos fijos.

B) Posicionamiento hiperbólico

En la figura 6.11 se muestra el mismo tipo de test pero en este caso empleando el método LZC (método 1) y el método GN hiperbólico. Se puede observar que en este caso las diferencias entre ambos métodos son mucho mayores que en el caso esférico: en valores medios absolutos en torno a los 30 mm en el eje 'X' y sobre 15 mm en el eje 'Y'. Esto es debido a que aunque se supone que las balizas del LPS ultrasónico están a la misma altura, en la práctica esto no es exacto debido a que se emplea un reflector cónico, haciendo que dependiendo de la posición del móvil la altura efectiva a la que se encuentra cada baliza varíe, por eso hay puntos donde la diferencia es muy elevada y en cambio en otros esa diferencia es mucho menor.

Por último, en la figura 6.12 se puede ver la comparativa empleando en este caso el método LTI (método 3) y el algoritmo GN hiperbólico. En este caso al ser un método que no necesita que las balizas se encuentren a la misma altura, los resultados son mucho mas parecidos, en torno a 2 mm de error absoluto exceptuando los puntos donde se produce el fenómeno del multitrayecto, en esta zona la diferencia aumenta hasta los 15 mm.

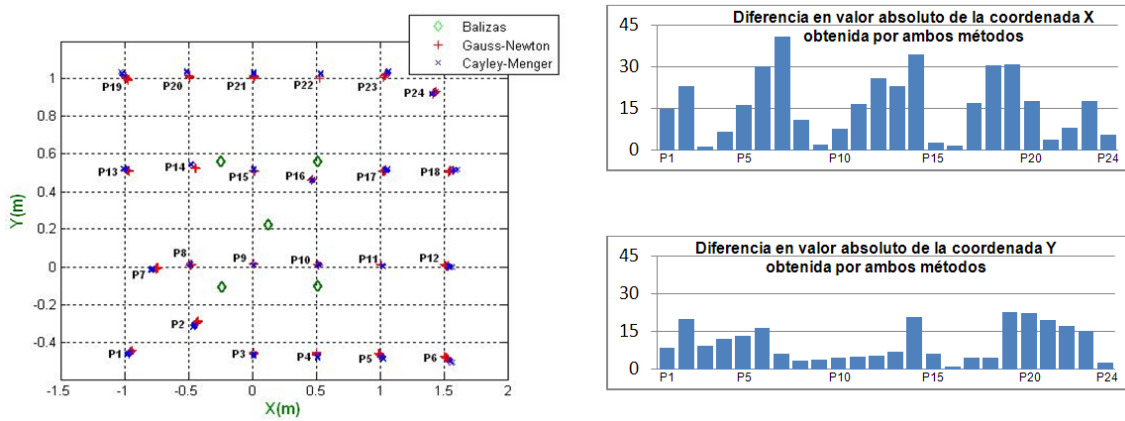


Figura 6.11: Comparativa entre el método LZW y el método GN hiperbólico para una rejilla de puntos fijos.

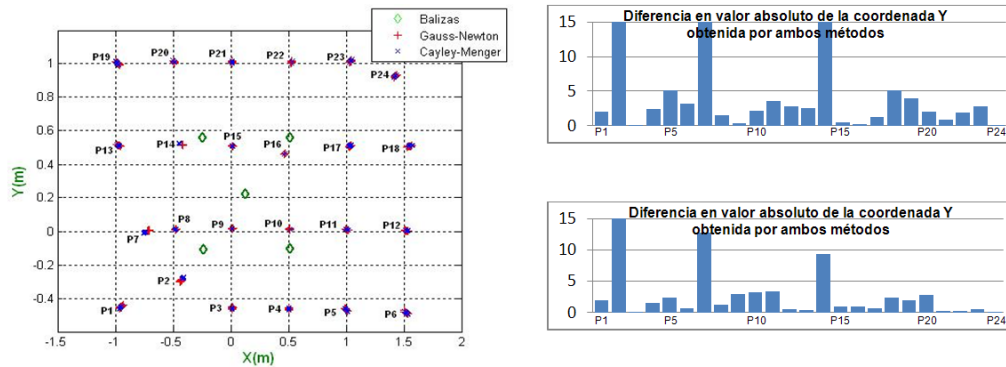


Figura 6.12: Comparativa entre el método LTI y el algoritmo GN hiperbólico para una rejilla de puntos fijos.

6.2.2. Pruebas en movimiento

En estas pruebas se vuelven a comparar los mismos métodos que en las pruebas de puntos fijos, pero esta vez realizando un recorrido con el robot móvil dentro del área de cobertura.

A) Posicionamiento esférico

En la figura 6.13 se pueden observar tanto las posiciones obtenidas con el algoritmo GN esférico como el algoritmo esférico basado en los bideterminantes de Cayley-Menger. Se puede comprobar visualmente que las posiciones son prácticamente las mismas, de hecho la diferencia media entre ambos algoritmos es de 0,74 mm en el eje 'X' y de 0.94 mm en el eje 'Y' con unas diferencias máximas de 2.8 mm en la componente 'X' y de 7.5 mm en la componente 'Y'. En la gráfica también se pueden observar zonas de errores sistemáticos debidos a efectos del multicamino.

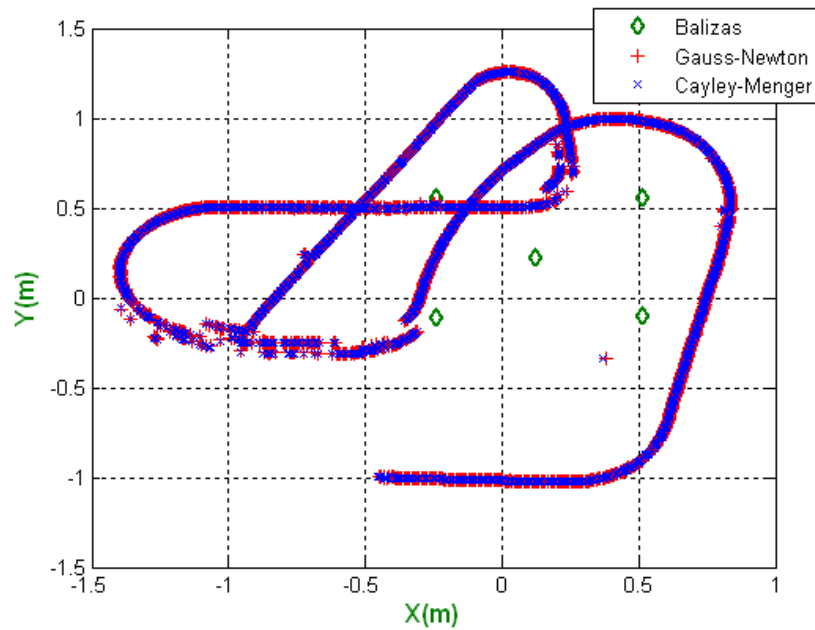


Figura 6.13: Comparativa entre el algoritmo basado en GN esférico y el algoritmo esférico basado en los bideterminantes de Cayley-Menger, realizando un recorrido dentro del área de cobertura.

En la figura 6.14 se muestra el valor absoluto de la diferencia en cada uno de los puntos del recorrido para las coordenadas X e Y. Se puede observar cómo en el caso en que los errores en la medida son muy pequeños, la diferencia entre ambos algoritmos es siempre inferior a 1.5 mm y sólo en la zona donde se producen multitrayectos, esto es en donde las medidas contienen un mayor error la diferencia entre los dos métodos aumenta.

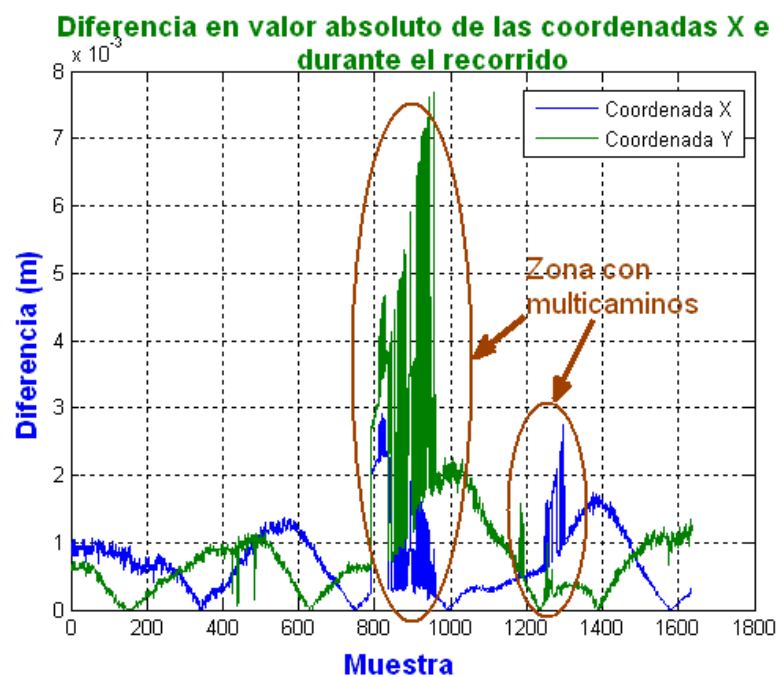


Figura 6.14: Diferencia en valor absoluto de las coordenadas X e Y obtenida con el método GN esférico y el algoritmo esférico basado en los bideterminantes de Cayley-Menger.

B) Posicionamiento Hiperbólico

En la figura 6.15 se muestra el mismo recorrido, solo que ahora se emplean el algoritmo propuesto en el método LZC (método 1) y el algoritmo de resolución GN hiperbólico. Como ocurrió en la prueba de puntos fijos, al no ser la altura real de las balizas constante sino que varía en función de la posición debido al reflector, la diferencia entre ambos métodos es mayor. En este caso el error medio cometido en los ejes X e Y es de 11.4 mm y de 10.8 mm respectivamente, mientras que los errores máximos son de 50.4 mm y 49.8 mm .

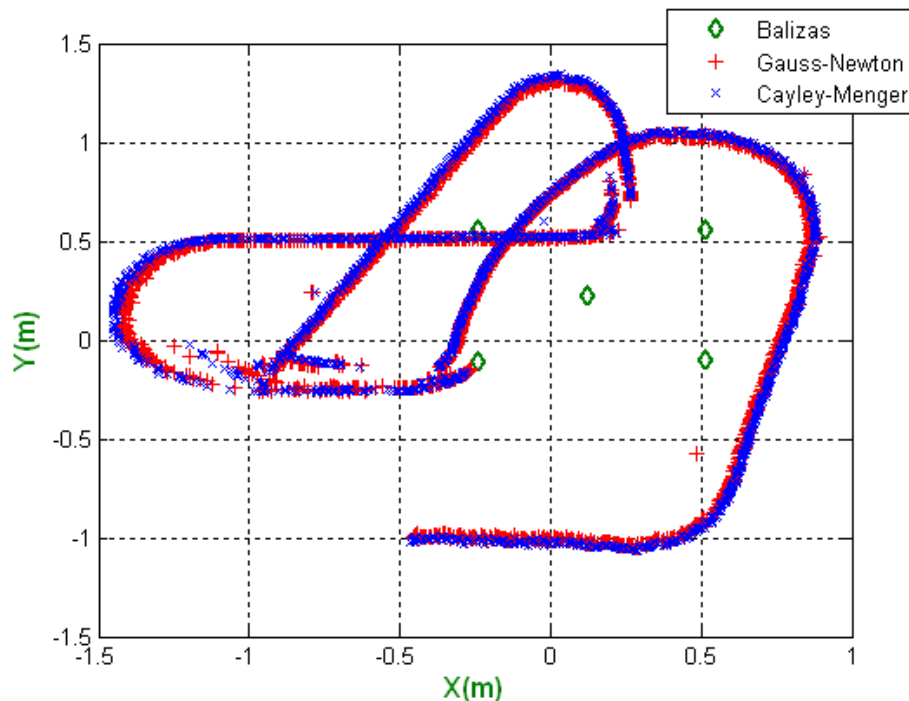


Figura 6.15: Comparativa entre el algoritmo basado en Gauss-Newton y el método LZC basado en Cayley-Menger que supone que las balizas están en un plano Z constante, realizando un recorrido dentro del área de cobertura.

En la figura 6.16 se vuelve a mostrar el valor absoluto de la diferencia entre el método LZC y el método GN hiperbólico en todos los puntos del recorrido. Se comprueba que esta diferencia es mayor que en el caso esférico por los motivos mencionados anteriormente. También se comprueba que en las zonas donde existe multicamino la diferencia entre ambos métodos se incrementa hasta 5 cm , pero exceptuando estas zonas el valor de la diferencia entre ambos métodos es inferior a 2 cm .

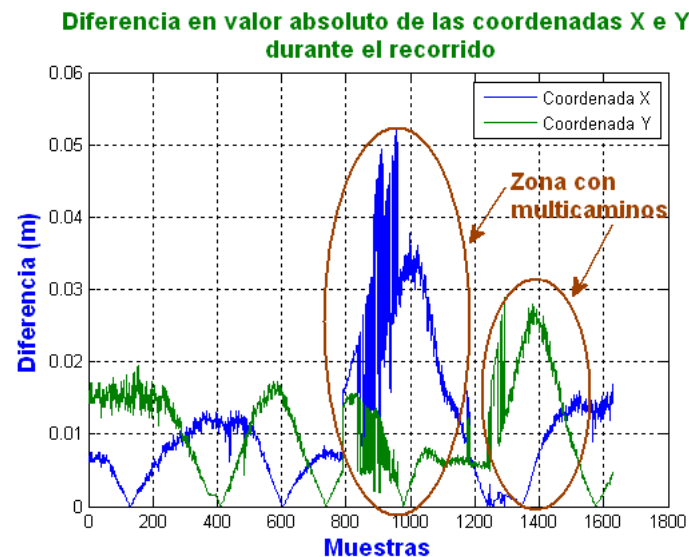


Figura 6.16: Diferencia en valor absoluto de las coordenadas X e Y obtenida con el método LZC (todas las balizas a la misma altura) y el método GN hiperbólico.

En la figura 6.17 se puede observar el mismo recorrido pero esta vez se compara el algoritmo presentado en el método LTI y el método GN hiperbólico. En este caso sí que se observa una diferencia importante y es que el algoritmo propuesto es capaz de filtrar algunos “outlayers” que se producen en las medidas. Esto es debido a que el método LTI incorpora una raíz cuadrada que cuando las medidas contienen mucho error se vuelve imaginaria permitiendo detectar de esa forma los “outlayers”. En las demás zonas se observa que la diferencia es mínima, de hecho la diferencia media es de 1.8 mm en el eje X y de 1.98 mm en el eje Y con unos errores máximos en torno a los 5 mm en ambos ejes.

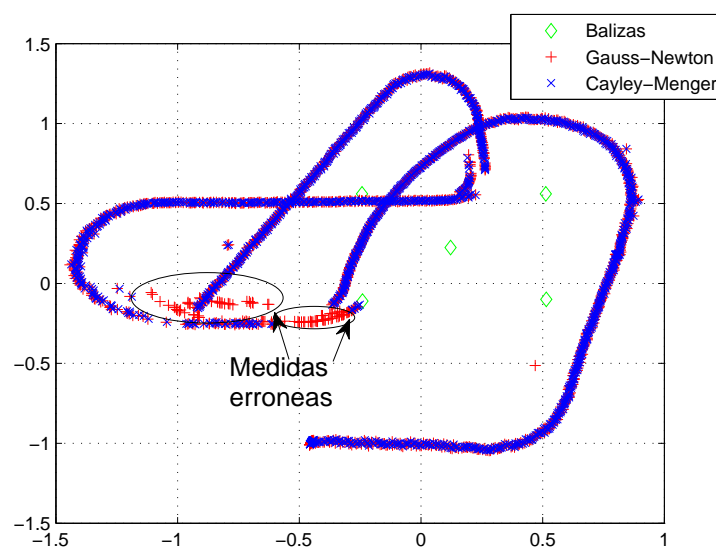


Figura 6.17: Comparativa entre el algoritmo basado en Gauss-Newton y el método LTI basado en Cayley-Menger, realizando un recorrido dentro del área de cobertura.

En la figura 6.18 se vuelve a mostrar el valor absoluto de la diferencia entre los dos métodos. Se puede observar cómo los resultados son muy similares, ya que el algoritmo no requiere que las balizas estén a la misma altura.

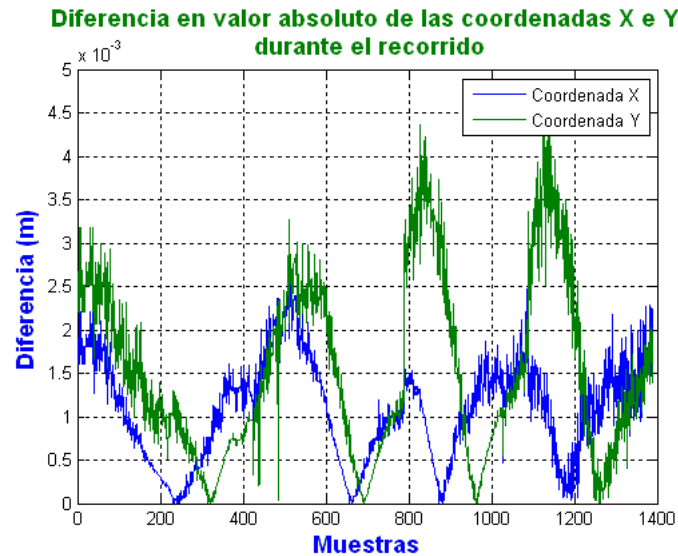


Figura 6.18: Diferencia en valor absoluto de las coordenadas X e Y obtenida con el método LTI (método iterativo) y el método GN hiperbólico.

En este caso no se han efectuado las pruebas de ejecución de velocidad debido a que el recorrido del robot no es muy largo, unos 1.543 puntos, y las posiciones se han calculado de modo offline. Pero en una implementación en tiempo real, los algoritmos propuestos requieren una carga computacional menor, que el método GN.

6.3. Pruebas de autocalibración empleando medidas desde puntos desconocidos

Para poder comprobar la precisión de los algoritmos propuestos, fué necesario realizar una calibración manual del sistema. La realización de esta calibración fue tediosa y requirió un importante esfuerzo y cuidado. Los pasos para dicha calibración fueron los siguientes:

- Con un medidor láser que permitía trazar líneas perpendiculares se trazó en el área de cobertura una rejilla de puntos separados 0.5 metros entre sí y se fijó un origen de referencia (véase la figura 6.19).
- Mediante una plomada se obtuvo la proyección de las balizas en el suelo.
- Se calculó la posición de las balizas al origen de referencia.
- Por último se midió la altura de las balizas respecto al suelo.

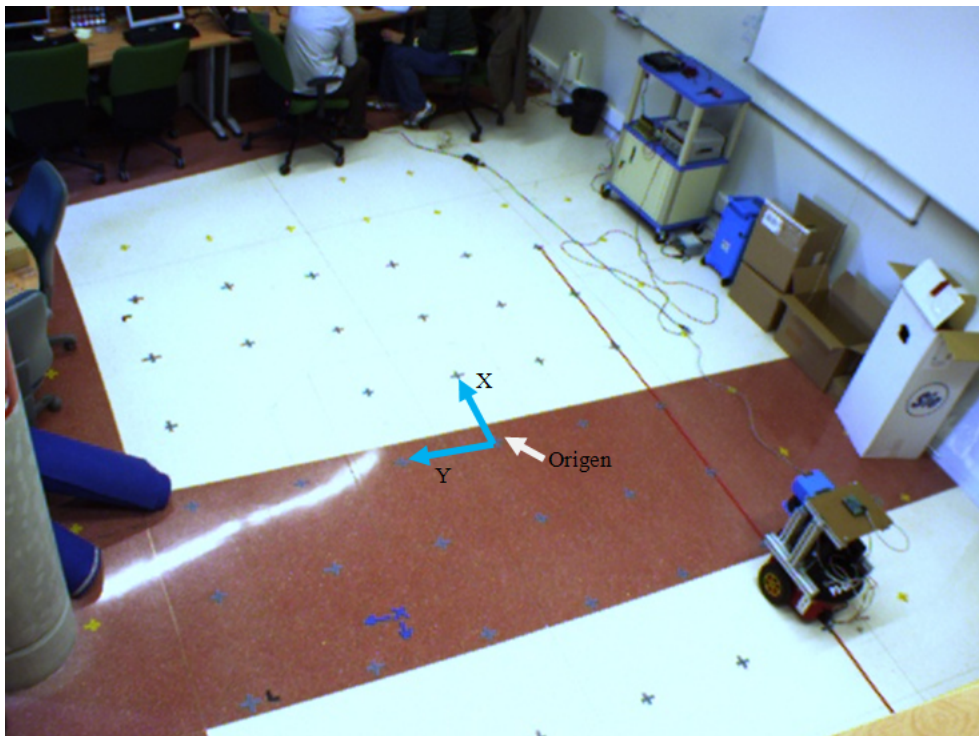


Figura 6.19: Rejilla para realizar la calibración dentro del área de cobertura.

Una vez realizado todo este proceso la posición de las balizas respecto al origen de coordenadas se muestra en la tabla 6.2

Tabla 6.2: Posición de las balizas después de la calibración manual.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.122 m.	0.513 m.	0.515 m.	-0.24 m.	-0.242 m.
Coordenada Y	0.224 m.	0.56 m.	-0.1 m.	-0.11 m.	0.56 m.
Coordenada Z	3.5 m.	3.5 m.	3.5 m.	3.5 m.	3.5 m.

En una primera serie de tests se prueban los algoritmos para el posicionamiento esférico para a continuación realizar las pruebas para el posicionamiento hiperbólico. Para todas las pruebas se supondrá que la posición de los puntos de test conocidos es la que se muestra en la tabla 6.3.

Tabla 6.3: Posiciones $[x, y, z]$ de los puntos de test conocidos para las pruebas.

Punto 1	Punto 2	Punto 3
$[0, 0, 0.03]$ m.	$[0.5, 0, 0.03]$ m.	$[0, 0.5, 0.03]$ m.

6.3.1. Pruebas para posicionamiento esférico

En este apartado se realizan varias pruebas de calibración empleando trilateración esférica, esto es hay sincronización entre las balizas y el receptor. En una primera parte se realizan pruebas con cuatro puntos de test desconocidos y en una segunda parte se emplean 8 puntos de test desconocidos

6.3.1.1. Partiendo de cuatro puntos de test desconocidos

Para la realización de esta prueba se toman 20 medidas, tanto en los 3 puntos de test conocidos como en los 4 puntos de test desconocidos, cuyas posiciones reales se muestran en la tabla 6.4. En este caso y en todos los que siguen los puntos de test desconocidos se han ubicado en las zonas óptimas de acuerdo con el método propuesto en el capítulo 4.

Tabla 6.4: Posiciones $[x, y, z]$ de los 4 puntos de test desconocidos para la prueba de autocalibración esférica.

Punto 1	Punto 2	Punto 3	Punto 4
$[-1.5, -1, 0.03]$ m.	$[-1.5, 1.5, 0.03]$ m.	$[0.5, 1.5, 0.03]$ m.	$[1.5, 0.5, 0.03]$ m.

Una vez obtenidas las medidas se realiza la calibración del sistema empleando la versión esférica que se muestra en el apartado 4.2.1. Para cada una de las 20 medidas tomadas en todos los puntos de test, se realiza la calibración y la posición final de las balizas es la media de cada una de las posiciones calculadas. Obteniéndose las posiciones de las balizas mostradas en la tabla 6.5.

Tabla 6.5: Posición de las balizas después de la autocalibración esférica 3D con 4 puntos de test desconocidos.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.135 m.	0.507 m.	0.551 m.	-0.205 m.	-0.229 m.
Coordenada Y	0.207 m.	0.556 m.	-0.145 m.	-0.122 m.	0.541 m.
Coordenada Z	3.488 m.	3.495 m.	3.481 m.	3.492 m.	3.498 m.

En la figura 6.20 se muestra el error absoluto medio (para las 20 medidas) en centímetros para cada una de las componentes (x, y, z) de las balizas obtenidas mediante calibración. Se puede observar que, excepto para la baliza 3, el error cometido en cada componente es inferior a 2 cm. También se puede observar que la componente en la que el error es menor es la z .

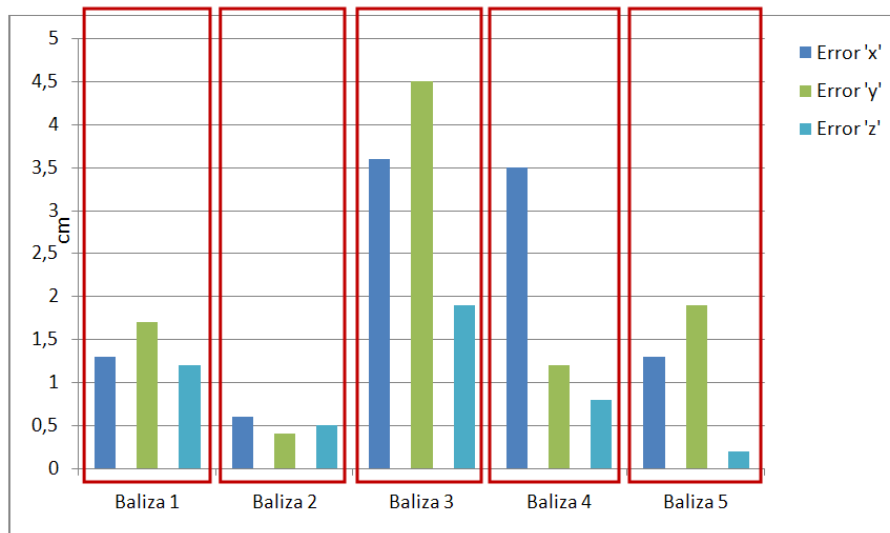


Figura 6.20: Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración esférica 3D con 4 puntos de test desconocidos y su posición real.

Por último se calcula el error medio de distancia en la calibración como la distancia media que existe entre la posición calibrada de las N balizas ($bc_{x,i}, bc_{y,i}, bc_{z,i}$) con su posición real ($b_{x,i}, b_{y,i}, b_{z,i}$) con $i = 1..N$:

$$e_m = \frac{1}{N} \sum_{i=1}^N \sqrt{(b_{x,i} - bc_{x,i})^2 + (b_{y,i} - bc_{y,i})^2 + (b_{z,i} - bc_{z,i})^2}$$

Este error medio de distancia entre la posición de las balizas reales y las autocalibradas es de 3.10 cm.

Para comprobar como afecta el error cometido en la estimación de las balizas a la ubicación del móvil se realiza un recorrido con el robot dentro del área de cobertura y se comparan los resultados de posición del robot obtenidos tanto empleando la posición real de las balizas como la posición obtenida mediante calibración. El plano de medida del robot se sitúa a $z=0.60\text{ m}$ (altura del robot).

En la figura 6.21 se muestra el recorrido realizado por el robot, tanto si se emplea las balizas reales (azul) como las calibradas (rojo). Como la localización del robot se realiza en 3D, para facilitar la visualización de resultados se ha dividido la representación en 2 partes, en (a) se muestra el posicionamiento del mismo en 2D, mientras que en (b) se muestra el plano $X-Z$, pudiendo analizarse de esta forma la altura obtenida para el plano de medida. Se puede observar que los dos recorridos son muy similares y el error medio de distancias cometido empleando las balizas calibradas respecto a las reales es de 4.41 cm con un error máximo de 6.99 cm y una desviación típica de 1.68 cm.

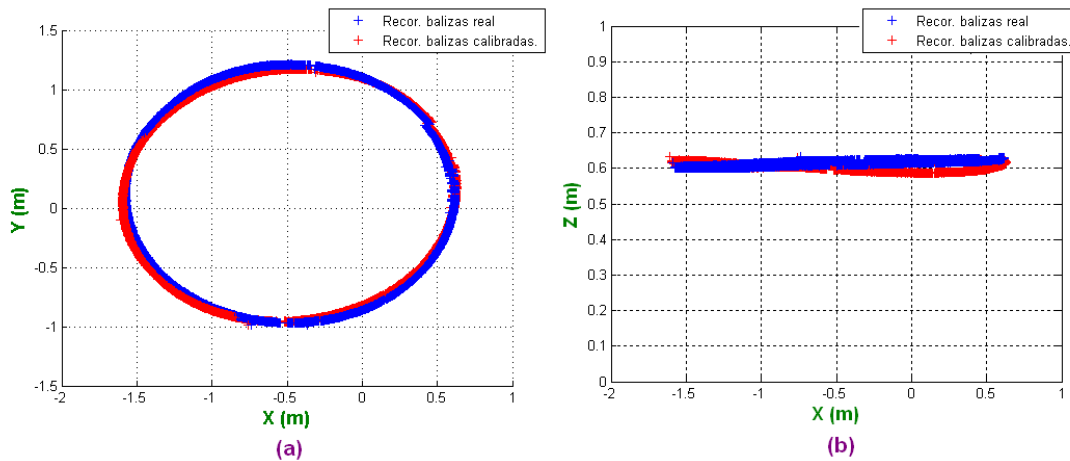


Figura 6.21: Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 3D, empleando 4 puntos de test desconocidos y posicionamiento esférico.

Si en el sistema de posicionamiento local implementado las balizas se encuentran a la misma altura (cosa bastante habitual ya que se suelen situar en el techo de una habitación) a la hora de realizar la calibración manual, si bien es difícil obtener sus componentes X e Y , es mucho más sencillo obtener la componente Z ya que sólo es necesario medir la altura al suelo. En el caso de poder obtener el valor de esa componente sólo es necesario realizar la calibración 2D de las balizas, en la tabla 6.6 se muestra la posición de las balizas obtenidas si se realiza una calibración 2D y los mismos 4 puntos de test desconocidos empleados en el test anterior.

Tabla 6.6: Posición de las balizas después de la autocalibración esférica 2D con 4 puntos de test desconocidos.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.121	0.502	0.532	-0.221	-0.239
Coordenada Y	0.226	0.562	-0.103	-0.103	0.537
Coordenada Z	3.5	3.5	3.5	3.5	3.5

En la figura 6.22 se vuelve a mostrar el error absoluto medio cometido en cada una de las componentes (x, y) de las 5 balizas obtenidas en la nueva calibración, no se muestra la componente z porque se supone que es conocida. Se puede observar que los errores son menores ya que en casi todos los casos la diferencia es menor a 2 cm. Además, en este caso, el error medio de distancia en la localización de las balizas pasa de los 3.10 cm del caso anterior a 1.47 cm, consiguiéndose una reducción de más del 50 %.

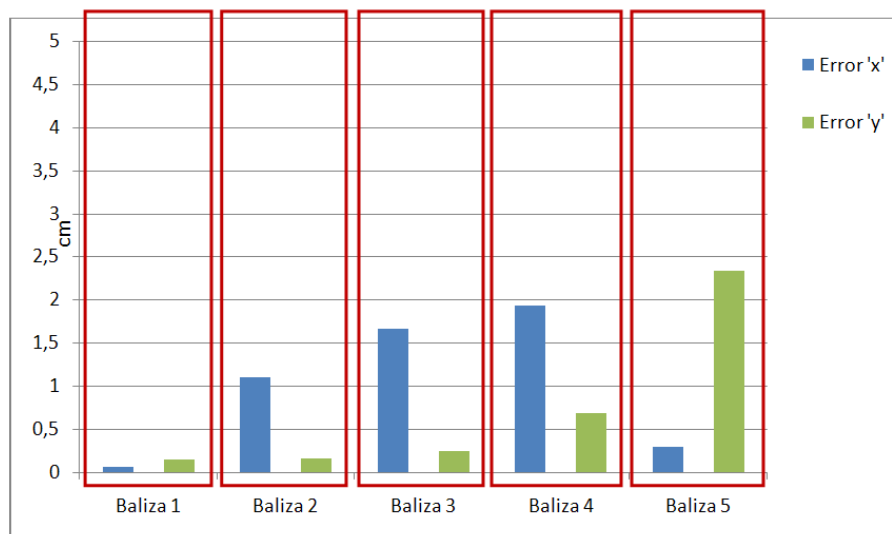


Figura 6.22: Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante auto-calibración esférica 2D con 4 puntos de test desconocidos y su posición real.

Por último en la figura 6.23 se muestran los resultados obtenidos cuando el robot realiza el recorrido de prueba tanto si se emplea la posición real de las balizas, como si se emplea la posición calibrada. En este caso los resultados de posición se acercan algo más a los obtenidos empleando las posiciones reales de las balizas. El error medio de todas las posiciones se reduce a 2.73 cm con un error máximo de 5.33 cm y una desviación típica de 1.07 cm. De lo que se puede apreciar una mejora en la estimación de la posición del móvil.

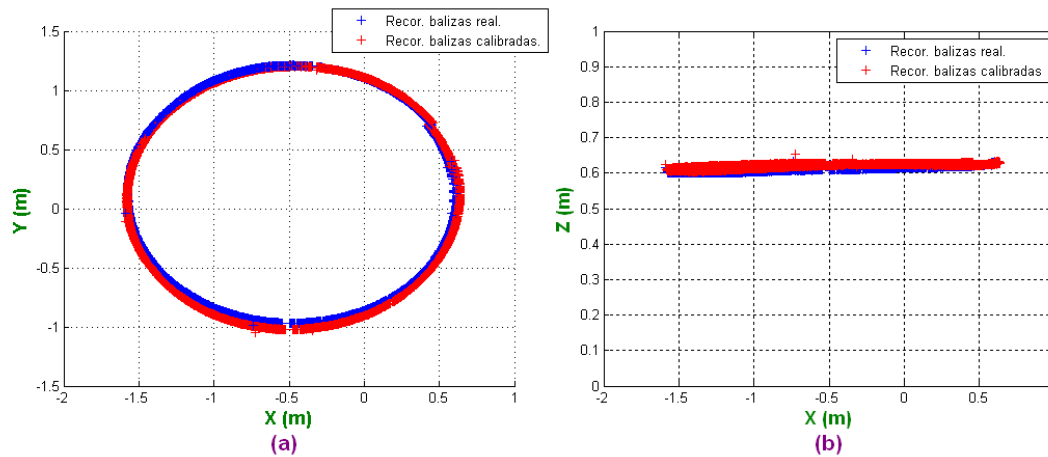


Figura 6.23: Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 2D, empleando 4 puntos de test desconocidos y posicionamiento esférico.

6.3.1.2. Partiendo de ocho puntos de test desconocidos

En esta sección se muestran los resultados obtenidos para diferentes pruebas reales de calibración empleando ocho puntos de test desconocidos cuya posición real es la indicada en la tabla 6.7

Tabla 6.7: Posición de los 8 puntos de test desconocidos para la realización de las pruebas de calibración esférica.

	Pto 1	Pto 2	Pto 3	Pto 4	Pto 5	Pto 6	Pto 7	Pto 8
Coord. X	0.0	-1.5	2.0	-1.5	0.5	1.5	2.0	-1.5
Coord. Y	-1.5	-1.5	0.5	1.5	1.5	1.5	-1.5	0
Coord. Z	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03

En la primera prueba se realiza la calibración en tres dimensiones de la posición de las balizas y en la tabla 6.8 se muestra la localización de las balizas obtenidas.

Tabla 6.8: Posición de las balizas después de la autocalibración esférica 3D con 8 puntos de test desconocidos.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.122	0.504	0.542	-0.234	-0.242
Coordenada Y	0.212	0.561	-0.109	-0.119	0.534
Coordenada Z	3.487	3.495	3.488	3.489	3.496

En la figura 6.24 se muestran los errores absolutos medios en cada una de las componentes (x, y, z) en la posición de las balizas, se puede observar una mejora respecto a la

misma calibración realizada con sólo 4 puntos de test. En este caso el error medio cometido es de 2.04 cm, reduciéndose el error comparado con el caso de autocalibración 3D empleando 4 puntos de test desconocidos (3.1 cm), pero mayor que si la calibración se realiza en 2D (1.47 cm)

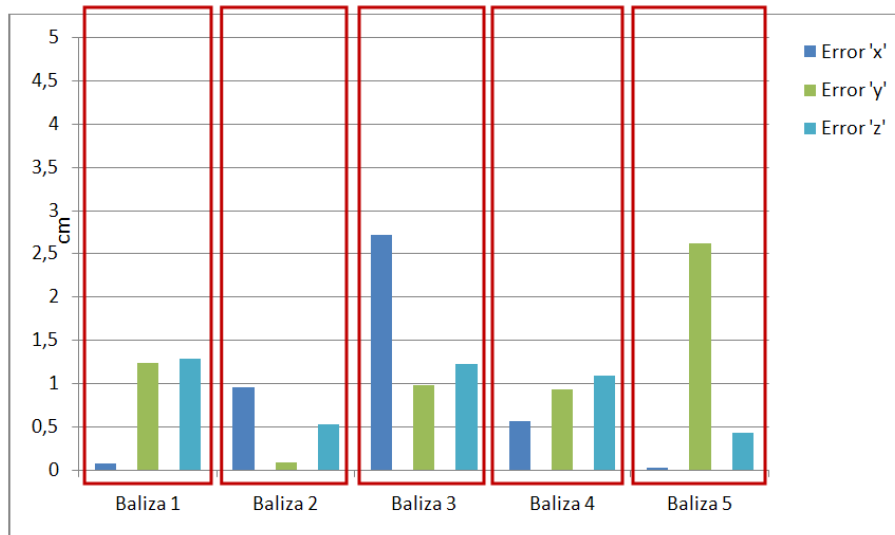


Figura 6.24: Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante auto-calibración esférica 3D con 8 puntos de test desconocidos y su posición real.

A continuación en la figura 6.25 se muestran las posiciones obtenidas en el recorrido de prueba si se emplea las posiciones de las balizas reales (azul) o si se emplean las posiciones de las balizas estimadas (rojo). Se comprueba visualmente que las dos trayectorias son muy similares, siendo el error medio absoluto en distancia en la trayectoria de 3.20 cm, con un error máximo de 5.43 cm y una desviación típica de 1.37 cm.

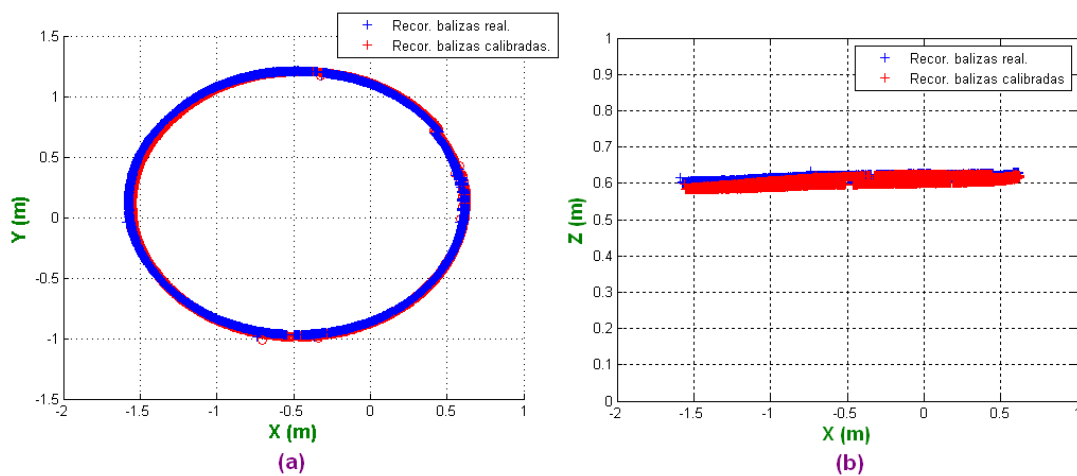


Figura 6.25: Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 3D, empleando 8 puntos de test desconocidos y posicionamiento esférico.

La última prueba consiste en realizar la misma autocalibración, pero esta vez suponiendo que se conoce la altura de las mismas y sólo es necesario obtener las componentes X e Y de las balizas. La ubicación de los 8 puntos de test es la misma que en la prueba anterior. La posición de las balizas calibradas obtenidas se muestra en la tabla 6.9.

Tabla 6.9: Posición de las balizas después de la autocalibración esférica 3D con 8 puntos de test desconocidos.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.125	0.509	0.540	-0.231	-0.240
Coordenada Y	0.217	0.571	-0.104	-0.114	0.545
Coordenada Z	3.5	3.5	3.5	3.5	3.5

Como en el resto de las pruebas en la tabla 6.26 se muestran los errores absolutos medios cometidos en cada una de las componentes x e y de la posición de las balizas. Se observa que para casi todos los casos el error es inferior a 1 cm; el único caso donde el error es más elevado es en la componente x de la baliza 3 donde el error llega a los 2.5 cm. En este caso el error medio de distancias cometido es el menor de todos ya que se reduce a 1.39 cm

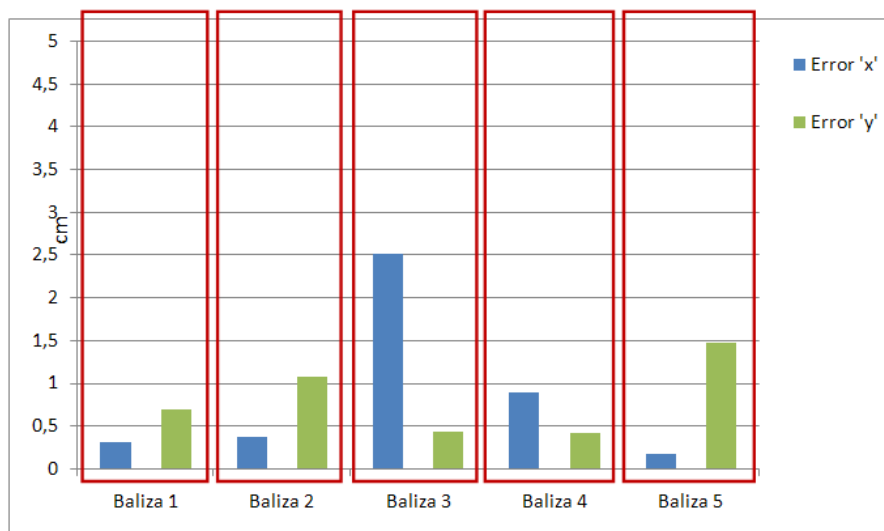


Figura 6.26: Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante autocalibración esférica 2D con 8 puntos de test desconocidos y su posición real.

Por último en la figura 6.27 se muestra como en los casos anteriores las posiciones obtenidas empleando tanto la posición real de las balizas como la calibrada, observándose que las posiciones calculadas son muy similares. En esta prueba el error medio absoluto en distancia en las posiciones es de 2.57 cm, con un error máximo de 5.25 cm y una

desviación típica de 1.02 cm. Se comprueba que, aumentando el número de puntos de test desconocidos y realizando una calibración 2D, mejoran los resultados obtenidos.

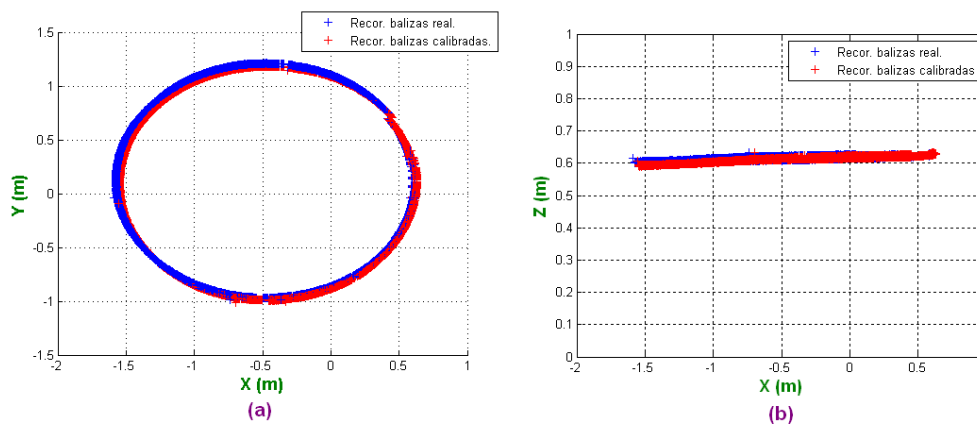


Figura 6.27: Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 3D, empleando 8 puntos de test desconocidos y posicionamiento esférico.

6.3.2. Pruebas para posicionamiento hiperbólico

En esta sección se presentan los resultados reales obtenidos cuando en vez de emplear un posicionamiento esférico, se emplea un posicionamiento hiperbólico. Esto implica que no es necesaria la sincronización entre las balizas y el receptor.

En este caso, el PDOP del sistema real es muy elevado, varía entre 50 y 85 (figura 6.28 a) por lo que aunque los errores en la medida de tiempo de vuelo son muy bajos, no ha sido posible realizar una calibración 3D del mismo, ni siquiera colocando los puntos de test desconocidos en las zonas óptimas.

Analizando más en detalle el PDOP, esto es descomponiéndolo en sus componentes horizontal (HDOP, figura 6.28 b) y vertical (VDOP, figura 6.28 c) se observa que los errores más elevados son debidos al VDOP, ya que éste varía entre 50 y 70, mientras que el HDOP se mantiene en valores más reducidos (10 y 40), por lo que si bien no es posible realizar una calibración 3D, si que es más factible realizar una calibración en 2D ya que en este caso la componente vertical es conocida (se mide a mano) y el sistema no está afectado por el VDOP.

Como en las pruebas de posicionamiento esférico se realizan test para 4 y 8 puntos de test desconocidos.

6.3.2.1. Partiendo de cuatro puntos de test desconocidos

En este caso se calcularon las zonas óptimas y se situaron los puntos de test desconocidos en las posiciones que indica la tabla 6.10. La posición de los puntos de test conocidos es la misma que en los anteriores casos, la indicada en la tabla 6.3.

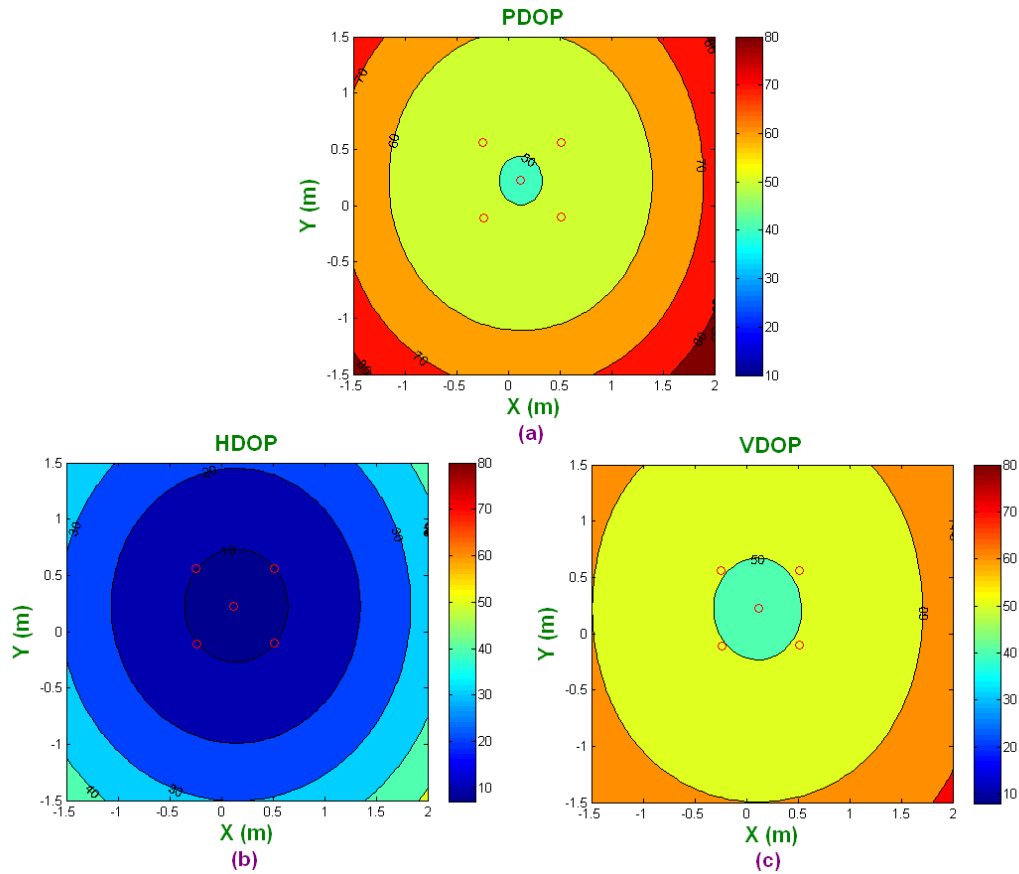


Figura 6.28: PDOP del sistema del posicionamiento local cuando se emplea posicionamiento hiperbólico dentro de la zona de cobertura. (a) PDOP hiperbólico del sistema. (b) HDOP hiperbólico del sistema. (c) VDOP hiperbólico del sistema

Tabla 6.10: Posiciones $[x, y, z]$ de los 4 puntos de test desconocidos para la prueba de autocalibración hiperbólica.

	Punto 1	Punto 2	Punto 3	Punto 4
Coordenada X	1	1	-1.5	-1.5
Coordenada Y	-0.5	1.0	-0.5	1.0
Coordenada Z	0.03	0.03	0.03	0.03

Después del proceso de calibración las posiciones obtenidas de las balizas se muestran en la tabla 6.11.

Tabla 6.11: Posición de las balizas después de la autocalibración hiperbólica 2D con 4 puntos de test desconocidos.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.094	0.442	0.491	-0.229	-0.274
Coordenada Y	0.233	0.564	-0.079	-0.070	0.554
Coordenada Z	3.5	3.5	3.5	3.5	3.5

En la figura 6.29 se muestran los errores absolutos medios cometidos en cada componente. Se puede ver que son más elevados que los calculados en cualquier prueba de posicionamiento esférico. En este caso el error medio cometido es de 4.1 cm, siendo el error más elevado de todos los casos.

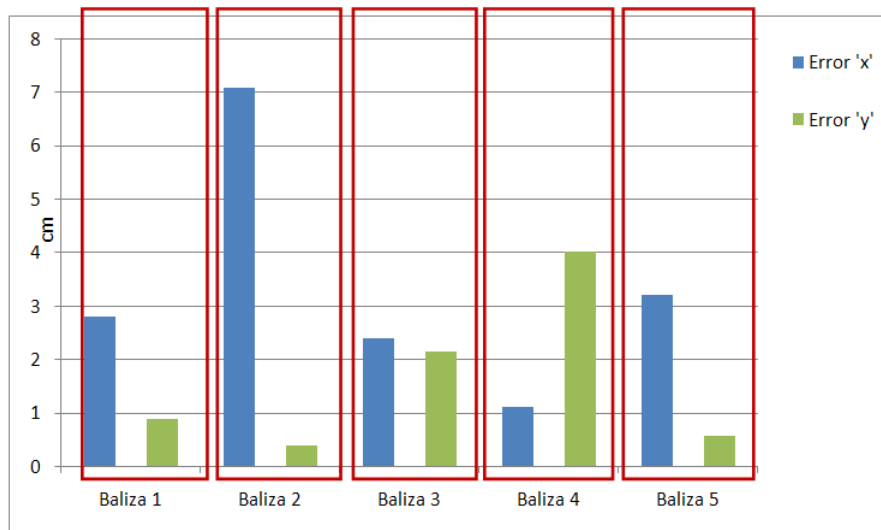


Figura 6.29: Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante auto-calibración hiperbólica 2D con 4 puntos de test desconocidos y su posición real.

Se vuelve a realizar el mismo recorrido en el área de cobertura; en la figura 6.30 se muestran las posiciones obtenidas tanto si se emplean las posiciones reales como si se emplean las posiciones estimadas. En este caso, a la hora de posicionar se emplea la función de posicionamiento hiperbólico basada en los determinantes de Cayley-Menger, y debido al alto valor del VDOP se realiza un posicionamiento en 2 dimensiones, por lo que la altura del plano de medida es conocida (0.60 cm). En este caso se observa que los errores en distancia cometidos son mucho más elevados, sobre todo en la zona inferior izquierda. En este recorrido el error absoluto medio en distancia en las posiciones es de 10.33 cm, mucho más elevado que en las anteriores ocasiones, con un error máximo de 21.45 cm y una desviación típica de 6.45 cm.

6.3.2.2. Partiendo de ocho puntos de test desconocidos

A continuación se realiza el proceso de calibración hiperbólico en 2D pero empleando esta vez 8 puntos de test desconocidos, cuya ubicación real se muestra en la tabla 6.12.

En la tabla 6.13 se muestra la posición de las balizas obtenidas. En este caso el error medio cometido es de 3.21 cm. Este error es más elevado que los obtenidos en el caso esférico pero menor que el obtenido con la calibración hiperbólica con 4 puntos de test desconocidos. Además en la figura 6.31 se muestran los errores absolutos medios en cada

una de las componentes. Se aprecia que el error se reduce comparado con el caso anterior, sobre todo el error máximo que en este caso es de 4.5 cm y en el caso anterior fue superior a 7 cm.

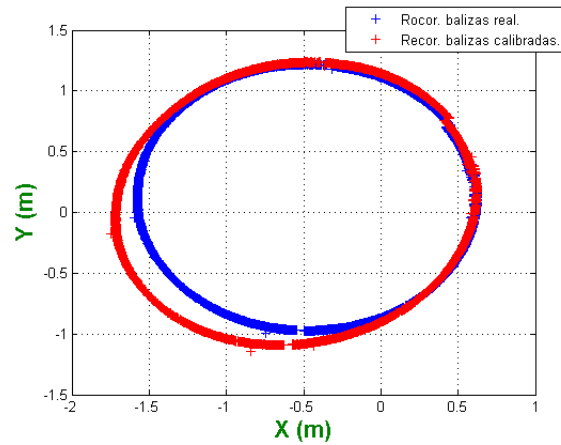


Figura 6.30: Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 2D, empleando 4 puntos de test desconocidos y posicionamiento hiperbólico.

Tabla 6.12: Posición de los 8 puntos de test desconocidos para la realización de las pruebas de calibración hiperbólica.

	Pto 1	Pto 2	Pto 3	Pto 4	Pto 5	Pto 6	Pto 7	Pto 8
Coord. X	0	-1.5	-1.5	-1.5	0.5	1.5	2.0	-1.5
Coord. Y	-1.5	-1.5	0.0	1.5	1.5	1.5	-1.5	0
Coord. Z	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03

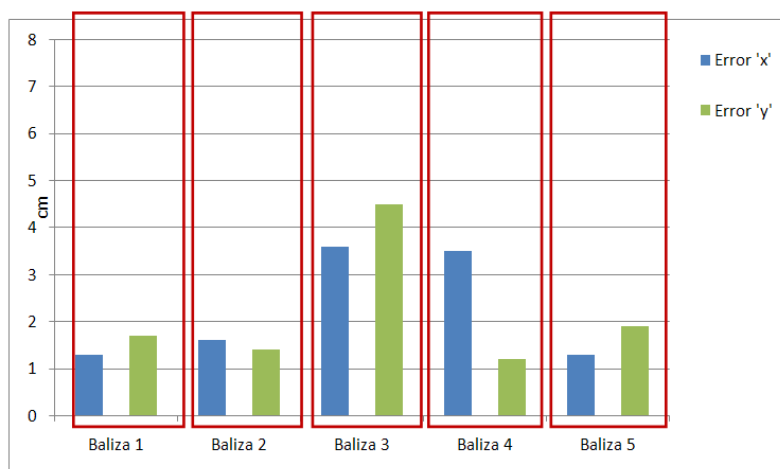


Figura 6.31: Error absoluto medio en centímetros entre la posición de las balizas obtenidas mediante auto-calibración hiperbólica 2D con 8 puntos de test desconocidos y su posición real.

Tabla 6.13: Posición de las balizas después de la autocalibración hiperbólica 2D con 8 puntos de test desconocidos.

	Baliza 1	Baliza 2	Baliza 3	Baliza 4	Baliza 5
Coordenada X	0.135	0.497	0.551	-0.205	-0.229
Coordenada Y	0.207	0.546	-0.145	-0.122	0.541
Coordenada Z	3.5	3.5	3.5	3.5	3.5

Por último en la figura 6.32 se muestra las posiciones obtenidas con el recorrido de prueba. Se puede observar cómo en este caso la posición estimada del móvil mediante las balizas calibradas se acerca más a la posición obtenida con la posición real de las balizas. En esta prueba el error absoluto medio en distancias en la posición es de 7.27 cm con un error máximo de 12.43 cm y una desviación típica de 3.87 cm, confirmandose la mejora obtenida en los resultados al emplear los 8 puntos de test desconocidos para realizar la calibración.

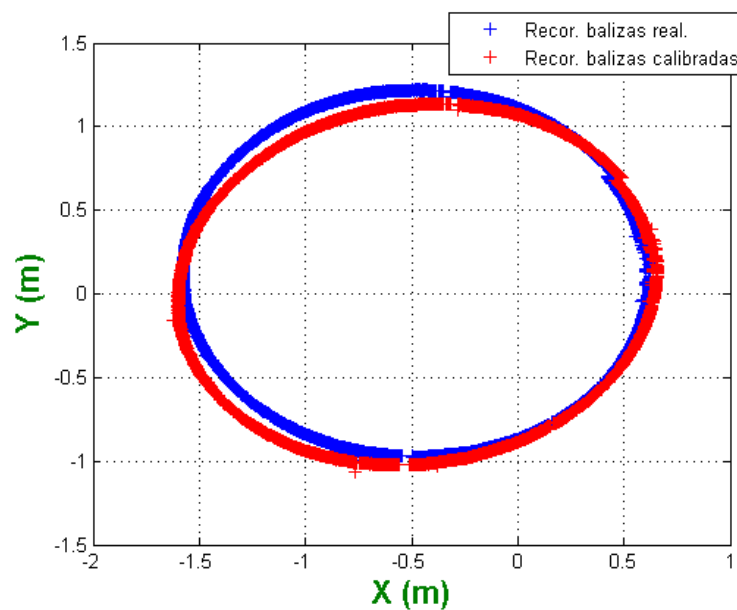


Figura 6.32: Comparación en el recorrido realizado por el robot tanto si se emplea la posición de las balizas reales, como si se emplea la posición obtenida mediante calibración 2D, empleando 8 puntos de test desconocidos y posicionamiento hiperbólico.

6.4. Pruebas de localización y calibración simultáneas empleando el filtro $H-\infty$

En esta sección se muestran los resultados reales obtenidos en el LPS descrito en las anteriores pruebas cuando se emplea el algoritmo de autocalibración y posicionamiento simultáneos basados en el filtro $H-\infty$. Como se ha mencionado en el capítulo 5, donde se explica este método, cuando se realiza la calibración se fija como origen de referencia la posición inicial del robot y como eje X la dirección a la que está apuntado. Como es muy complicado hacer coincidir el eje de referencia empleado en la calibración manual con el eje de referencia empleado en la autocalibración para que las posiciones de las balizas obtenidas sean las mismas, lo que se hace en esta prueba es comparar la estructura geométrica de las balizas, esto es la distancia entre las balizas (figura 6.33), con la estructura obtenida con la autocalibración.

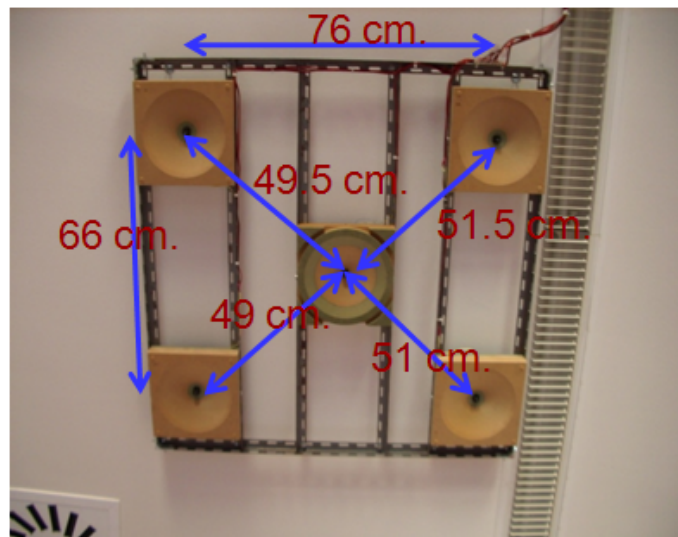


Figura 6.33: Estructura de las balizas del LPS empleado en las pruebas reales. Además todas las balizas se encuentran a 3.5 m del suelo.

En la figura 6.34 se muestra el recorrido realizado con el robot dentro del área de cobertura del sistema LPS, los círculos negros representan las posiciones obtenidas con la odometría, las cruces azules las posiciones estimadas y además se han marcado las posiciones empleadas para realizar la precalibración.

Los resultados obtenidos, se exponen en la figura 6.35. Se muestran las estructuras obtenidas tanto después de la precalibración como después de la fase de calibración, se puede observar que las distancias en el segundo caso se aproximan mucho más a las distancias reales. Para evaluar esta mejora se calcula empíricamente el error medio absoluto de distancia cometido en la estimación, siendo en el caso de la precalibración de 2.50 cm y después de la fase de calibración se reduce a 1.35 cm.

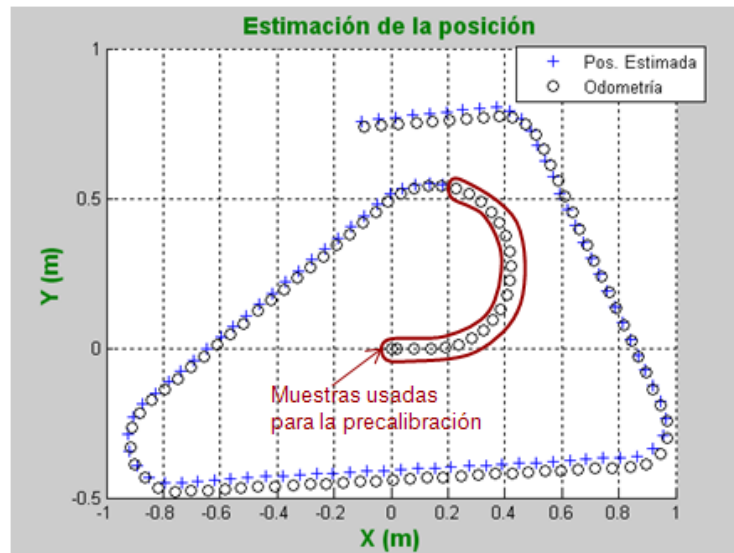


Figura 6.34: Recorrido realizado con el robot para realizar la calibración con el método basado en el filtro H_∞ . Además se emplean las primeras muestras para realizar la precalibración.

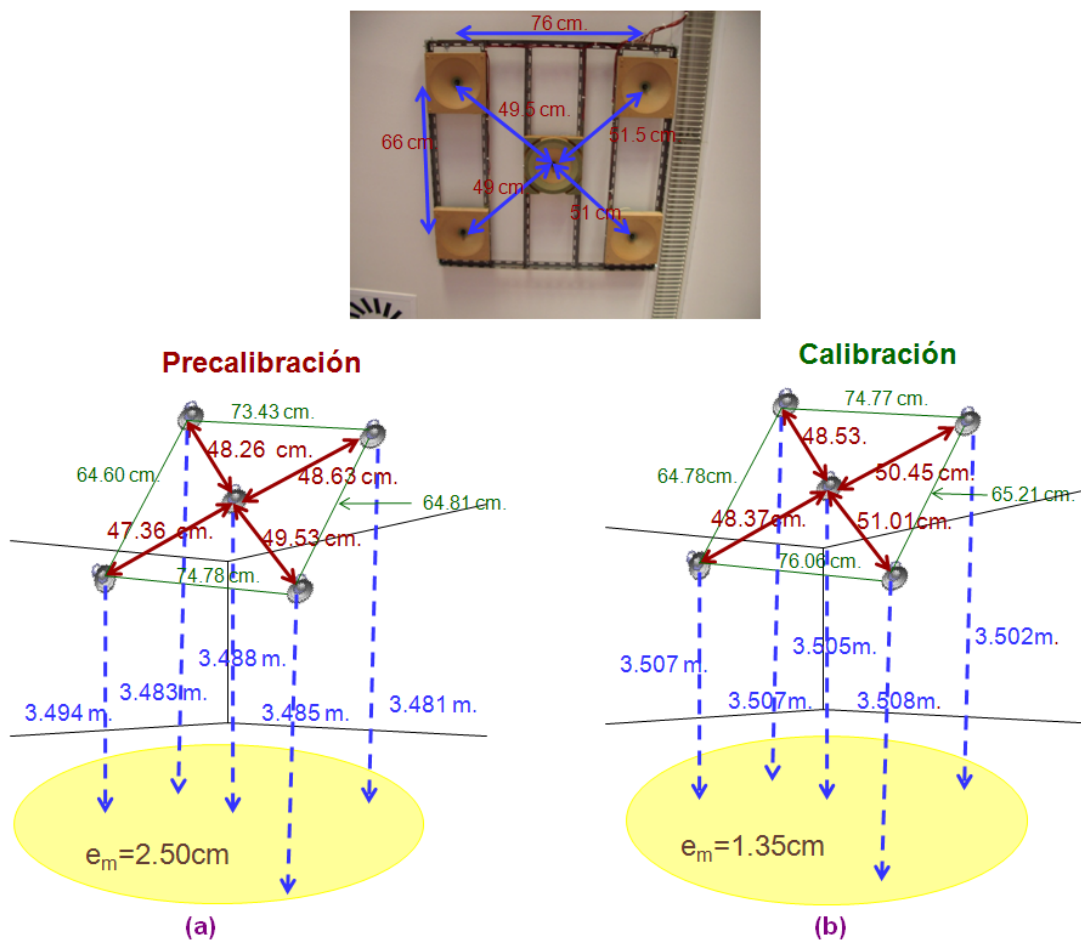


Figura 6.35: Resultados obtenidos después de realizar la autocalibración mediante el método basado en el filtro H_∞ . (a) Estructura de las balizas después del proceso de precalibración. (b) Resultados obtenidos después de la calibración. La altura real de las balizas es 3.5 m.

En la siguiente prueba se vuelve a realizar una precalibración del sistema pero esta vez se introducen dos deslizamientos de aproximadamente 1 segundo en la odometría del robot para comprobar el funcionamiento del algoritmo en esta situación. En la figura 6.36 se muestra el recorrido realizado por el robot en esta prueba. Se vuelven a mostrar tanto las posiciones de la odometría como la posición estimada por el algoritmo. En este caso se puede observar la zona donde se produce el deslizamiento y cómo a partir de ese punto la trayectoria estimada por el robot no coincide con la odometría.

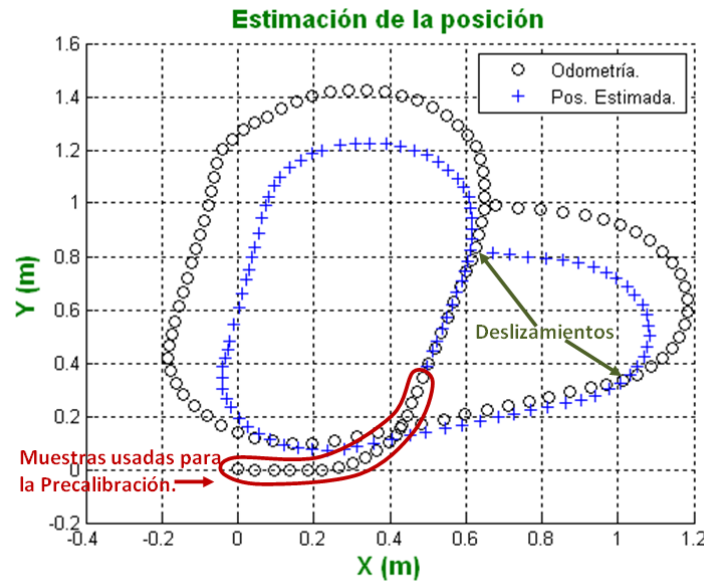


Figura 6.36: Recorrido realizado con el robot para realizar la calibración con el método basado en el filtro $H-\infty$. Se producen dos deslizamientos durante la fase de Calibración. Además se emplean las primeras muestras para realizar la precalibración.

En la figura 6.37 se muestran las diferentes estructuras de balizas obtenidas en las pruebas, se puede comprobar, que a pesar de los deslizamientos, la etapa de calibración consigue disminuir el error medio absoluto en distancia cometido en la fase de Precalibración, ya que el error original de 3.91 cm se reduce a 2.18 cm, aunque el error final es algo más elevado que cuando no se producen deslizamientos durante la fase de Calibración.

6.5. Conclusiones

En este capítulo se han presentado los resultados obtenidos en un sistema de posicionamiento local real de los diferentes algoritmos propuestos en esta tesis. En una primera sección se ha descrito el sistema LPS que se ha utilizado para realizar las pruebas y se han analizado sus componentes. A continuación se han realizado sobre el mismo una serie de pruebas con cada uno de los métodos propuestos y se han podido extraer las siguientes conclusiones:

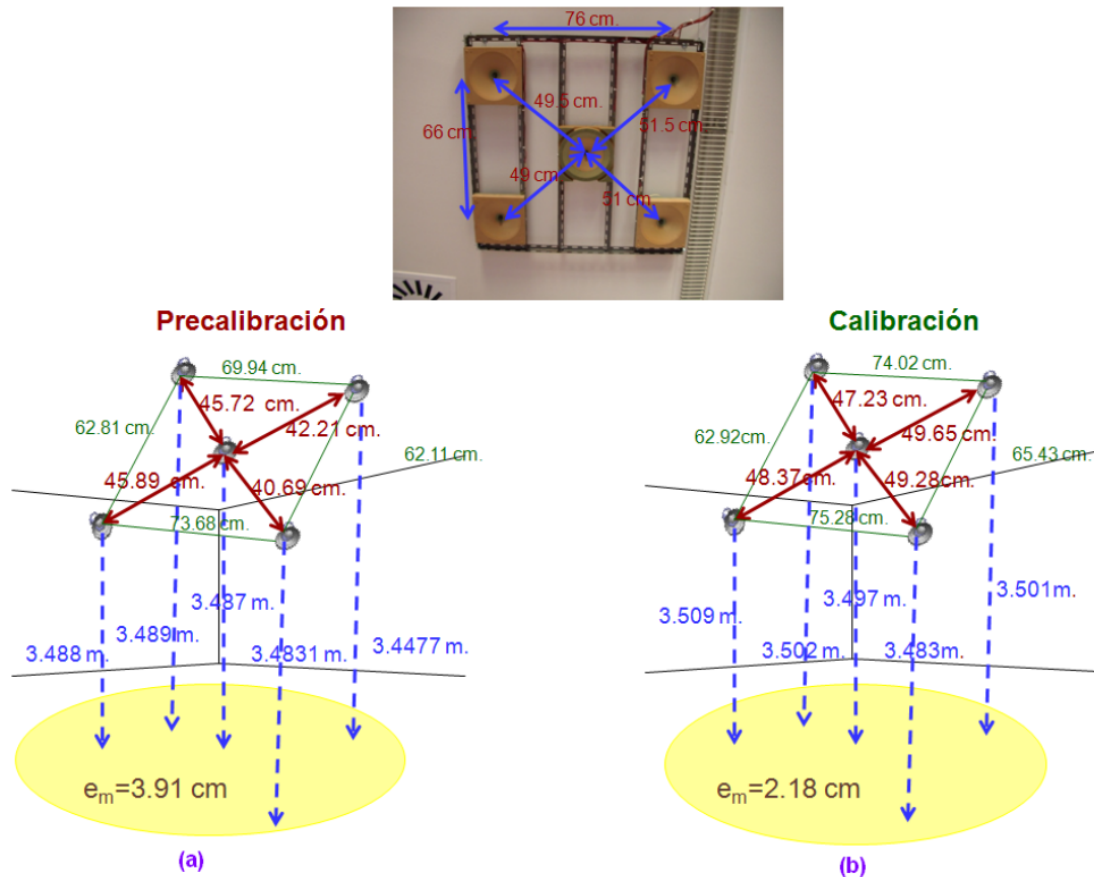


Figura 6.37: Resultados obtenidos después de realizar la autocalibración mediante el método basado en el filtro H- ∞ . Se introducen deslizamientos en la odometría del robot en la fase de calibración. (a) Estructura de las balizas después del proceso de precalibración. (b) Resultados obtenidos después de la calibración. La altura real de las balizas es 3.5 m.

■ **Algoritmos de posicionamiento basados en los bideterminantes de Cayley-Menger:**

Para los diferentes métodos de posicionamiento propuestos en esta tesis se han realizado pruebas tanto en puntos fijos como pruebas en movimiento, comparando los resultados con el algoritmo de resolución directa empleando Gauss-Newton (GN) [YTBJ08], de los resultados obtenidos se puede concluir:

- En el caso del método LTI (método 3) la precisión obtenida cuando se posiciona objetos es muy similar a la obtenida con el GN hiperbólico, ya que la diferencia entre ambos es menor de 5 mm en la mayor parte de los casos. Además se ha comprobado en las pruebas de movimiento que el algoritmo propuesto es capaz de discriminar “outlayers” en las medidas debido a que cuando las medidas son muy erróneas hacen que este método proporcione posiciones imaginarias.
- En el caso del algoritmo propuesto en el método LZC (método 1). Se ha comprobado que los resultados no son tan similares como se obtuvo en las pruebas bajo simulación, en las pruebas reales la diferencia entre ambos métodos es

inferior a 3 cm en la mayor parte de los casos. Este incremento se debe a que si bien se ha supuesto que las balizas se encuentran a la misma altura, esto no ocurre ya que el reflector cónico del LPS hace que la altura efectiva de las balizas varíe según la ubicación del nodo móvil. En las pruebas en movimiento se han obtenido unos resultados similares a los de las pruebas en puntos fijos.

- **Pruebas de Autocalibración empleando medidas en puntos desconocidos:** Se han realizado pruebas tanto para un posicionamiento esférico, como para un posicionamiento hiperbólico y además se ha comprobado como varían los resultados obtenidos si se aumenta el número de puntos de test en posiciones desconocidas. De los resultados obtenidos se pueden extraer las siguientes conclusiones:

- Se obtiene un resultado mejor en la autocalibración de las balizas con un posicionamiento esférico que con un posicionamiento hiperbólico. De hecho, el mejor resultado obtenido con el posicionamiento hiperbólico (un error de 3.21 cm en la posición de las balizas) comete más error que el peor resultado obtenido con el posicionamiento esférico (un error de 3.1 cm en la estimación de las balizas). Además, debido al elevado VDOP que presenta el LPS, en el caso hiperbólico no ha sido posible realizar una calibración 3D del sistema de balizas, siendo necesario medir la altura de las mismas y el calibrado posterior sólo en 2D.
- Se ha comprobado que tanto en el caso esférico como en el hiperbólico, al aumentar el número de puntos de test desconocidos en el sistema, aumenta la precisión en la autocalibración del mismo.
- Por último también se ha comprobado que si es posible medir la altura del sistema de balizas (algo que no es muy complicado) se puede realizar una calibración 2D del mismo, mejorándose de esa forma los resultados en la calibración.

- **Pruebas de localización y calibración empleando un filtro $H-\infty$:** Debido a que actualmente sólo hay disponible un sistema LPS, sólo se han realizado pruebas de calibración de un único LPS; simultáneamente al proceso de calibración se ha calculado la posición del robot. Además como la posición final de las balizas depende de la posición inicial del robot (esta posición marca el origen de referencia y su orientación) en las pruebas se han comparado las estructuras de las balizas obtenidas.

- Se ha comprobado que el método propuesto mejora las posiciones obtenidas en la etapa de precalibración, con una reducción del error en torno al 40%.

Además consigue simultáneamente obtener la posición del robot dentro del área de cobertura.

- En el caso de introducir deslizamientos en la odometría del robot, se ha comprobado que el algoritmo es robusto y consigue estimar la posición de las balizas. También se ha comprobado cómo la posición estimada consigue evitar los errores debidos a los deslizamientos gracias a la información recibida del LPS.

Conclusiones y trabajos futuros

En este capítulo se exponen las conclusiones más relevantes obtenidas de esta tesis, indicándose también las líneas futuras de investigación sobre la temática tratada. Además, se incluyen las publicaciones derivadas de este trabajo relacionándolas con los aspectos abordados.

7.1. Conclusiones

En esta tesis se ha analizado el problema de la calibración de un sistema de posicionamiento local, esto es, el obtener de forma automática la posición de las balizas, ya que esta información normalmente es necesaria para poder calcular la posición del nodo móvil. En este sentido las contribuciones más relevantes realizadas en la tesis son las que se detallan a continuación:

Nuevo algoritmo para posicionamiento hiperbólico

Se ha propuesto un algoritmo para trilateración hiperbólica derivado del método para la trilateración esférica propuesto por [TR05]. Este algoritmo presenta la ventaja de que no necesita de métodos de resolución iterativa complejos para obtener la posición del nodo móvil ya que, al utilizar las propiedades del operador de Cayley-Menger, consigue encontrar la posición de forma directa. En una primera aproximación se han modificado las ecuaciones para adaptarlas para el caso hiperbólico y posteriormente, analizando las nuevas ecuaciones no lineales, se han propuesto tres nuevos métodos de resolución:

- El primer método propuesto (LZC) es empleable si se cumple la restricción geométrica de que todas las balizas se encuentran en un mismo plano de Z constante, esto es, que estén a la misma altura. Esta situación es bastante común, ya que todas las balizas se suelen situar en el techo de la habitación. Bajo estas condiciones se eliminan las no linealidades de las ecuaciones y se puede encontrar la posición del nodo móvil resolviendo un sistema lineal de ecuaciones. Este método, en las pruebas realizadas, demuestra ser mucho más rápido que el método de resolución directa de las ecuaciones de distancia de Gauss-Newton dando resultados con similar precisión: menos de 5 *mm* de diferencia en las coordenadas calculadas en las

pruebas simuladas. En las pruebas reales presenta unas diferencias en torno a los 3 cm, debido a que el LPS real presenta un reflector cónico que hace que la altura efectiva del sistema no sea la misma para todas las balizas.

- El segundo método (LTR) es válido para cualquier distribución de las balizas y a través de una rotación del sistema de referencia, consigue que al menos tres balizas sí estén en el mismo plano convirtiendo de esa forma el sistema de ecuaciones no lineal en un sistema cuadrático. En las pruebas realizadas se ha comprobado la eficiencia computacional del método pero también se ha constatado que para determinadas configuraciones puede ofrecer una precisión peor que los demás algoritmos.
- Por último se ha propuesto un tercer método de resolución de las ecuaciones de modo iterativo (LTI). En los diferentes test se ha verificado que este método es algo más lento que los dos anteriores pero más rápido que el método de resolución directa mediante Gauss-Newton y que ofrece una precisión similar, tanto en las pruebas bajo simulación como en las pruebas reales, con diferencias en las coordenadas calculadas inferiores a 5 mm.

Diseño de un método de calibración para un LPS individual

Existen diversos algoritmos que tratan de obtener la posición de las balizas del LPS empleando un sistema de posicionamiento inverso, esto es, se toman medidas desde varios puntos conocidos y se obtiene la posición de las balizas. En esta tesis se ha presentado un método para realizar la calibración de un sistema LPS balizado, tanto si se emplea posicionamiento esférico como hiperbólico, que no necesita conocer la ubicación de todos los puntos de medida para encontrar las posiciones de las balizas. De esos puntos de medida, el método propuesto sólo necesita conocer la ubicación de tres, mientras que del resto no es necesario que se conozca su posición. Además, también se propone un método que permite conocer las mejores zonas óptimas para ubicar los puntos de test desconocidos.

- En el caso esférico, se ha visto en las pruebas bajo simulación que los resultados obtenidos para una correcta distribución de balizas, esto es con un bajo PDOP en el área de cobertura, mejoran ligeramente ubicando los puntos de medida en las zonas óptimas sobre todo para desviaciones típicas de ruido en las medidas superiores a 3 cm. Esta mejora es más notable en el caso de que la distribución de balizas sea muy mala (elevado PDOP).
- Además se ha comprobado, tanto en las pruebas reales como simuladas, que en el caso esférico el aumento de los puntos de medida desconocidos en las zonas óptimas consigue que mejoren los resultados en las autocalibraciones.

- Para el caso hiperbólico se ha comprobado que, si se quiere realizar una autocalibración, es muy recomendable ubicar los puntos de medida en las zonas óptimas ya que si no los errores son demasiado elevados. Por ejemplo, para una distribución óptima de balizas y con una desviación típica de ruido en las medidas de 1 cm, el error puede reducirse en torno al 30 %, mientras que en las pruebas con una distribución peor de balizas, el error en la autocalibración puede reducirse en torno al 50 %.
- Al realizar simulaciones para el caso hiperbólico aumentando el número de puntos de medidas se ha comprobado que en el caso de que la distribución de balizas es muy mala, apenas se produce mejora, mientras que si la distribución de balizas es buena, sí que se consigue una mejora en los resultados.
- A la hora de comparar los resultados obtenidos en el caso esférico y en el caso hiperbólico se ha comprobado, como era de esperar, que los resultados obtenidos si se emplea un posicionamiento esférico son mucho mejores que si se emplea un posicionamiento hiperbólico. En las pruebas bajo simulación se ha comprobado que, para una buena distribución de balizas (esto es que tengan el menor PDOP posible en la zona de cobertura), el error medio cometido en el primer caso está en torno a 1 cm para desviaciones típicas de ruido en las medidas de 4 cm, mientras que en el caso hiperbólico sube hasta casi los 3 cm aún con desviaciones típicas en las medidas de 7 mm. Esto también ha quedado constatado en las pruebas reales ya que el mejor resultado obtenido con el posicionamiento hiperbólico (un error de 3.21 cm) en la posición de las balizas, comete más error que el peor resultado obtenido con el posicionamiento esférico (un error de 3.1 cm) en la estimación de las balizas. Además debido al elevado VDOP que presenta el LPS, en el caso hiperbólico no ha sido posible realizar una calibración 3D del sistema de balizas, siendo necesario medir manualmente la altura de las mismas.
- Por último en las pruebas reales se ha comprobado que, si es posible medir la altura del sistema de balizas algo que no es muy complicado con la distribución propuesta, se puede realizar una calibración 2D del mismo, mejorando de esa forma los resultados en la calibración.

Implementación de un método de calibración global para varios LPS independientes.

Otro gran problema que hay que considerar cuando se diseña un sistema LPS que quiera cubrir toda una planta de un edificio, es que el sistema puede estar compuesto por varios LPS independientes. En esta tesis se ha presentado un método de calibración global para varios LPS independientes usando un robot móvil. En una primera fase el algoritmo realiza la calibración del primero de los LPS mediante un filtro H- ∞ ; a continuación, una

vez finalizada la calibración, el robot móvil avanza hasta el siguiente LPS y realiza la calibración de este nuevo sistema respecto al sistema de referencia del primer LPS.

La fase de calibración de cada LPS independiente se divide en tres fases:

- Fase de Precalibración: donde a través de un recorrido del robot se realiza un posicionamiento inverso de las balizas.
- Fase de Calibración: mediante un filtro $H-\infty$ se fusionan los datos de odometría y la información del LPS para refinar la posición de las balizas calculadas en la fase de Precalibración.
- Fase de Navegación: una vez calibradas las balizas se adapta el filtro $H-\infty$ para navegar con el robot dentro del área de cobertura.

Además cuando el robot navega de un LPS calibrado a otro que no lo está, puede navegar por zonas donde no hay cobertura de ningún LPS. En estas zonas los errores introducidos por la odometría influyen gravemente en la calibración de los nuevos LPS. Por eso se ha propuesto un mecanismo para reducir dichos errores. De los resultados obtenidos se pueden extraer varias conclusiones:

- Se ha comprobado, tanto en pruebas bajo simulación como en pruebas experimentales, que la fase de calibración mejora notablemente los errores cometidos en la posición de las balizas por la fase de precalibración, consiguiendo que los errores originales se reduzcan en torno a 2 cm en las coordenadas X e Y ; en la componente Z el error se reduce a menos de 2 cm. Además se ha comprobado, tanto en pruebas bajo simulación como en pruebas reales, que el algoritmo es robusto a deslizamientos en las medidas de la odometría, ya que si bien justo en los momentos de los deslizamiento el error aumenta, si se sigue moviendo el robot dentro de la zona de cobertura el sistema se recupera de dicho error.
- Se ha comprobado bajo simulación que el método propuesto para reducir la influencia de los errores de la odometría en las zonas donde el robot no tiene apoyo de ningún LPS consigue reducir el error de calibración de forma significativa, ya que los errores originales con un orden de magnitud en torno a decímetros se reducen a errores en torno a centímetros. En estas zonas debido a la influencia del error de odometría se recomienda mover al robot lo más recto y despacio posible para evitar deslizamientos.
- Una vez realizada la calibración del sistema se ha comprobado el funcionamiento del filtro $H-\infty$ como método de guiado del robot dentro del sistema. Se ha probado su mejor funcionamiento y sus propiedades frente a otro sistema empleado típicamente en el guiado de robots móviles como el filtro de Kalman extendido.

7.2. Trabajos futuros

Esta tesis aborda la crucial tarea de la calibración en sistemas de posicionamiento basados en medidas de distancias a sus balizas. Como se demuestra en [TR05], el error cometido en la determinación de la posición de las balizas, afecta en la determinación del nodo móvil de igual forma que un error en las medidas de distancia; además la realización de este proceso de forma manual puede llevar gran cantidad de tiempo e introducir errores manuales. Los resultados presentados demuestran que se han conseguido mejoras gracias a los algoritmos propuestos; queda sin embargo mucho trabajo por hacer en este campo. Específicamente algunas de las tareas a realizar son:

- **Desarrollo y calibración de múltiples LPS**

En esta tesis no ha sido posible probar en un escenario real con múltiples LPS el algoritmo de calibración global basado en el filtro $H-\infty$. Actualmente se están desarrollando nuevos sistemas de posicionamiento ultrasónico con una gran área de cobertura pero sin la necesidad de emplear el reflector cónico para poder evitar de esta forma las incertidumbres en la posición z de las balizas.

- **Estudio de otras técnicas de fusión y navegación**

En el algoritmo de calibración global, el filtro $H-\infty$ fusiona la información procedente de la odometría y de los LPS. Se pueden estudiar como afectan a la calibración otros métodos de fusión como los Filtros de Partículas.

- **Estudio de nuevas técnicas de resolución de las ecuaciones no lineales**

En el algoritmo de trilateración hiperbólica basado en los bideterminantes de Cayley-Menger se ha llegado a una ecuación no lineal. Si las balizas se encuentran a la misma altura se puede linealizar el sistema, pero en el caso de que esto no ocurra, se necesita emplear un método iterativo para poder obtener la posición. Sería interesante plantear si se puede resolver el sistema mediante el empleo de las bases de Gröbner [For91], evitando de esa forma emplear el método iterativo.

- **Estudio de la ubicación de los puntos de test conocidos**

En el algoritmo de calibración para un sólo LPS, se ha propuesto un algoritmo para encontrar las zonas óptimas donde ubicar los puntos de test desconocidos. Se podría estudiar también como la ubicación de los puntos de test conocidos afecta a la calibración del sistema.

7.3. Publicaciones derivadas de la tesis

Publicaciones en revistas internacionales

- [RUG+11a] F. D. Ruiz, J. Ureña, J. C. García, J. M. Villadangos, M. Pérez, and E. García. Trilateration algorithm using differences of time of arrival. *Sensors and Actuators A: Physical*. En revisión.2011.

En este trabajo se expone el algoritmo de trilateración hiperbólica de la sección 3.3.2 y además se amplían pruebas reales y simuladas para los algoritmos de localización de las secciones 3.3.1 y 3.3.3.

Publicaciones en congresos internacionales

- [RUJ+08] F. D. Ruiz, J. Urena, J. A. Jimenez, J. M. Villadangos, J. J. Garcia, A. Hernandez, and A. Jimenez. Data processing for the calibration of an acoustic local positioning system. In *Proc. IEEE Instrumentation and Measurement Technology IMTC 2008*, pages 2085–2090, 2008.

En este trabajo se presentó un estudio de la influencia del número de puntos de test desconocidos empleados en la calibración en el error cometido en la posición de las balizas, tanto si se emplea un posicionamiento esférico como si se emplea un posicionamiento hiperbólico (vease la sección 4.4).

- [RUV+08] F. D. Ruiz, J. Urena, J. M. Villadangos, I. Gude, J. J. Garcia, A. Hernandez, and A. Jimenez. Optimal test-point positions for calibrating an ultrasonic lps system. In *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation ETFA 2008*, pages 338–344, 2008.

En este trabajo se presentó el algoritmo de determinación de zonas óptimas y un estudio comparativo de los resultados obtenidos si se situaban los puntos de test desconocidos de forma aleatoria o se situaban en dichas zonas óptimas de calibración (vease sección 4.3).

- [RUG+09b] D. Ruiz, J. Urena, I. Gude, J. M. Villadangos, J. C. Garcia, C. Perez, and E. Garcia. Hyperbolic ultrasonic LPS using a Cayley-Menger bideterminant-based algorithm. In *Proc. IEEE Instrumentation and Measurement Technology Conf. I2MTC '09*, pages 785–790, 2009.

En este trabajo se presentaron resultados, tanto reales como simulados, comparando el algoritmo de trilateración hiperbólica que supone que todas las balizas se encuentran a la misma altura basado en los bideterminantes de Cayley-Menger con el algoritmo de resolución directa de las ecuaciones de distancia empleando el método de resolución no

lineal de Gauss-Newton. Se analizaron tiempos de cómputo y precisión en las medidas (vease sección 3.3.1).

- [RUG+09a] D. Ruiz, J. Urena, I. Gude, J. M. Villadangos, J. C. Garcia, C. Perez, and E. Garcia. New iterative algorithm for hyperbolic positioning used in an ultrasonic local positioning system. In *Proc. IEEE Conf. Emerging Technologies & Factory Automation ETFA 2009*, pages 1–4, 2009.

En este trabajo se presentan los algoritmos de posicionamiento basados en los bideterminantes de Cayley-Menger descritos en las secciones 3.3.3 y una serie de pruebas bajo simulación de los mismos.

- [RUG+10a] D. Ruiz, J. Urena, J. C. Garcia, A. Hernandez, E. Garcia, and J. Aparicio. Simultaneous mobile robot positioning and LPS self-calibration in a smart space. In *Proc. IEEE Int Industrial Electronics (ISIE) Symp*, pages 2865–2870, 2010.

Se presenta el algoritmo de calibración basado en el filtro $H-\infty$ para un solo sistema LPS individual, se analizan sus prestaciones y sus propiedades (véase sección 5.3.2).

- [URG+10] Jesús Ureña, Daniel Ruiz, Juan Carlos García, Juan Jesus García, Enrique García. Ultrasonic lps: Autocalibration and mobile robot navigation. In *Abstracts of the 2010 International Conference on Indoor Positioning and Indoor Navigation*, 2010.

En este trabajo se presentan resultados bajo simulación del guiado de un robot móvil dentro de un LPS que ya ha sido calibrado mediante un filtro $H-\infty$. Además se comparan los resultados con el método de guiado empleando un filtro de Kalman Extendido (véase sección 5.1.3).

- [URG+11] J. Ureña, D. Ruiz, J. C. García, J. J. García, Á. Hernández, M. C. Pérez. LPS Self-Calibration Method using a Mobile Robot. In *Proc. IEEE Instrumentation and Measurement Technology Conf. I2MTC '11*, pages 1698–1703. Mayo 2011

En este trabajo se presentan pruebas, bajo simulación y reales, de la calibración global de un sistema LPS. Además se prueba el método de mejora de la estimación del segundo LPS (vease sección 5.2).

Publicaciones en congresos nacionales

- [RUV+08a] F. Daniel Ruiz, Jesús Ureña, José M. Villadangos, Isaac Gude, Juan J. García, Álvaro Hernández, Ana Jiménez. Optimización de la posición de los puntos de medida para calibración de un sistema LPS. *Actas del Seminario Anual de Automática, Electrónica Industrial e Instrumentación SAAEI*, 2008.

En este trabajo se realiza un análisis de las zonas óptimas en la ubicación de puntos óptimos, tanto para trilateración esférica como hiperbólica, empleando tanto resultados obtenidos en simulación como resultados reales (vease sección 4.4).

- [RUG+09c] D. Ruiz, J. Urena, I. Gude, J. M. Villadangos, J. C. Garcia, C. Perez, and E. Garcia. Posicionamiento Hiperbólico basado en el Bideterminante de Cayley-Menger. *Actas del Seminario Anual de Automática, Electrónica Industrial e Instrumentación SAAEI*, 2009.

En este trabajo se realiza una comparativa entre los resultados obtenidos con el algoritmo de posicionamiento basado en los determinantes de Cayley-Menger y el método de resolución directa de las ecuaciones mediante Gauss-Newton (vease sección 3.4).

- [RUG+10b] D. Ruiz, J. Urena, J. C. Garcia, A. Hernandez, E. Garcia, and J. Aparicio. Posicionamiento de un robot móvil y autocalibración de un LPS simultánea en espacios inteligentes. *Actas del Seminario Anual de Automática, Electrónica Industrial e Instrumentación SAAEI*, 2010.

Este trabajo presenta un analisis de los resultados cuando se realiza una calibración global de varios LPS independientes. Además se comprueba la robustez del algoritmo de calibración frente a deslizamientos en la etapa de calibración (vease sección 5.3).

- [RUG+11b] D. Ruiz, J. Urena, J. C. Garcia, J. J. García, S. Fernández, and D. Gualda. Calibración global de varios LPSs ultrasónicos usando un robot móvil. *Aceptado para publicación, SAAEI*, Julio 2011.

En este trabajo se presentan varias pruebas bajo simulación y reales para la calibración global de un sistema LPS. Además se muestran varias pruebas bajo simulación del método de mejora de la estimación del punto inicial de calibración del segundo LPS (vease sección 5.2).

Método de resolución no lineal de Gauss-Newton aplicado a trilateración

Este método encuentra la posición del nodo móvil mediante un proceso iterativo en el que se va llevando una posición que se estima del móvil, $\hat{\mathbf{p}} = [\hat{p}_x \hat{p}_y \hat{p}_z]$, a la posición real del mismo (figura A.1).

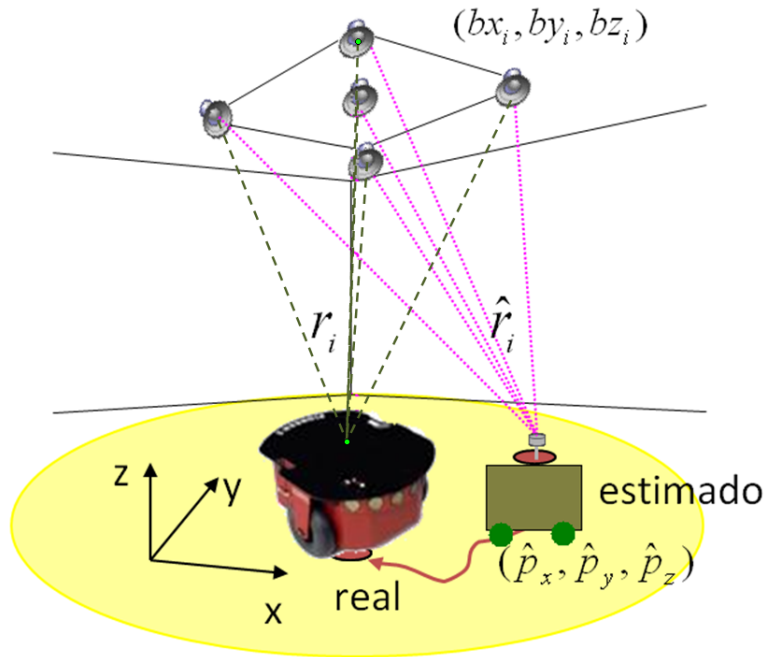


Figura A.1: Objetivo del método de resolución basado en Gauss-Newton.

A.1. Trilateración esférica

Si en un sistema LPS compuesto por N balizas, se define la posición de la baliza i como $\mathbf{b}_i = [bx_i \ by_i \ bz_i]$, entonces la distancia, que es la variable medida, entre la baliza \mathbf{b}_i y el punto a localizar, $\mathbf{p} = [p_x \ p_y \ p_z]$ puede definirse como:

$$d_i = \sqrt{(bx_i - p_x)^2 + (by_i - p_y)^2 + (bz_i - p_z)^2} \quad (\text{A.1})$$

De igual manera la distancia entre el punto estimado y las balizas puede expresarse como:

$$\hat{d}_i = \sqrt{(bx_i - \hat{p}_x)^2 + (by_i - \hat{p}_y)^2 + (bz_i - \hat{p}_z)^2} \quad (\text{A.2})$$

El método busca minimizar la suma de los errores cuadráticos de la distancia mediante la siguiente expresión:

$$F(p_x p_y p_z) = \sum_{i=1}^N (\hat{d}_i - d_i)^2 = \sum_{i=1}^N (f_i(p_x p_y p_z))^2 \quad (\text{A.3})$$

$$f_i(p_x p_y p_z) = (\hat{d}_i - d_i) = \sqrt{(bx_i - \hat{p}_x)^2 + (by_i - \hat{p}_y)^2 + (bz_i - \hat{p}_z)^2} - d_i \quad (\text{A.4})$$

Derivando $F(p_x p_y p_z)$ con respecto a \hat{p}_x , \hat{p}_y y \hat{p}_z se obtiene:

$$\frac{\partial F}{\partial \hat{p}_x} = 2 \sum_{i=1}^N f_i \frac{\partial f_i}{\partial \hat{p}_x}; \quad \frac{\partial F}{\partial \hat{p}_y} = 2 \sum_{i=1}^N f_i \frac{\partial f_i}{\partial \hat{p}_y}; \quad \frac{\partial F}{\partial \hat{p}_z} = 2 \sum_{i=1}^N f_i \frac{\partial f_i}{\partial \hat{p}_z} \quad (\text{A.5})$$

Reordenando el sistema para que quede de la forma $\mathbf{A} \cdot \Delta \mathbf{X} = \mathbf{B}$ se llega a:

$$\Delta \mathbf{X} = \begin{bmatrix} \Delta \hat{p}_x \\ \Delta \hat{p}_y \\ \Delta \hat{p}_z \end{bmatrix}; \quad \mathbf{A} = \begin{bmatrix} \frac{\partial f_1}{\partial \hat{p}_x} & \frac{\partial f_1}{\partial \hat{p}_y} & \frac{\partial f_1}{\partial \hat{p}_z} \\ \frac{\partial f_2}{\partial \hat{p}_x} & \frac{\partial f_2}{\partial \hat{p}_y} & \frac{\partial f_2}{\partial \hat{p}_z} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_N}{\partial \hat{p}_x} & \frac{\partial f_N}{\partial \hat{p}_y} & \frac{\partial f_N}{\partial \hat{p}_z} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} \quad (\text{A.6})$$

Donde:

$$\begin{aligned} \frac{\partial f_i}{\partial \hat{p}_x} &= \frac{(\hat{p}_x - bx_i)}{\hat{d}_i}; & \frac{\partial f_i}{\partial \hat{p}_y} &= \frac{(\hat{p}_y - by_i)}{\hat{d}_i}; & \frac{\partial f_i}{\partial \hat{p}_z} &= \frac{(\hat{p}_z - bz_i)}{\hat{d}_i}; & i &= 1 \dots N \\ f_i &= \hat{d}_i - d_i & i &= 1 \dots N \end{aligned} \quad (\text{A.7})$$

Resolviendo el sistema por mínimos cuadrados se obtiene que $\Delta \mathbf{X}$ es:

$$\Delta \mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B} \quad (\text{A.8})$$

Una vez se han obtenido los nuevos incrementos se calcula la nueva posición estimada del móvil como:

$$\hat{\mathbf{p}}_{k+1} = \hat{\mathbf{p}}_k - \Delta \mathbf{X}_k \quad (\text{siendo } k \text{ el orden de iteración}) \quad (\text{A.9})$$

Este proceso se repite hasta que los incrementos calculados sean inferiores a un umbral definido por el diseñador.

A.2. Trilateración hiperbólica

En este caso no se conocen las distancias absolutas entre el nodo móvil y las balizas, si no que se conoce el incremento de distancia entre el punto de medida con respecto a una baliza de referencia. En este desarrollo se va a suponer, sin pérdida de generalidad, que la baliza de referencia es la primera baliza.

Entonces, teniendo en cuenta lo anterior, el incremento de distancia de la baliza i situada en $\mathbf{b}_i = [bx_i \ by_i \ bz_i]$, respecto al punto $\mathbf{p} = [p_x \ p_y \ p_z]$, teniendo en cuenta que la baliza de referencia es la primera baliza, se puede expresar como:

$$\begin{aligned} \Delta d_{i,1} = d_i - d_1 &= \sqrt{(bx_i - p_x)^2 + (by_i - p_y)^2 + (bz_i - p_z)^2} \\ &- \sqrt{(bx_1 - p_x)^2 + (by_1 - p_y)^2 + (bz_1 - p_z)^2} \quad i = 2 \dots N \end{aligned} \quad (\text{A.10})$$

El incremento de distancia de esa misma baliza al punto estimado se puede expresar como:

$$\begin{aligned} \Delta \hat{d}_{i,1} = d_i - d_1 &= \sqrt{(bx_i - \hat{p}_x)^2 + (by_i - \hat{p}_y)^2 + (bz_i - \hat{p}_z)^2} \\ &- \sqrt{(bx_1 - \hat{p}_x)^2 + (by_1 - \hat{p}_y)^2 + (bz_1 - \hat{p}_z)^2} \quad i = 2 \dots N \end{aligned} \quad (\text{A.11})$$

Aplicando el proceso de la sección anterior, se busca minimizar el error cuadrático de la siguiente expresión:

$$F(p_x \ p_y \ p_z) = \sum_{i=1}^N (\Delta \hat{d}_i - \Delta d_i)^2 = \sum_{i=1}^N (f_i(p_x \ p_y \ p_z))^2 \quad (\text{A.12})$$

Al igual que en el caso anterior derivando y reordenando el sistema de la forma $\mathbf{A}\Delta\mathbf{X} = \mathbf{B}$ se llega a la expresión A.6, pero en este caso:

$$\begin{aligned}
 \frac{\partial f_i}{\partial \hat{p}_x} &= \frac{(\hat{p}_x - bx_i)}{\hat{d}_i} - \frac{(\hat{p}_x - bx_1)}{\hat{d}_1} \quad i = 2 \dots N \\
 \frac{\partial f_i}{\partial \hat{p}_y} &= \frac{(\hat{p}_y - by_i)}{\hat{d}_i} - \frac{(\hat{p}_y - by_1)}{\hat{d}_1} \quad i = 2 \dots N \\
 \frac{\partial f_i}{\partial \hat{p}_z} &= \frac{(\hat{p}_z - bz_i)}{\hat{d}_i} - \frac{(\hat{p}_z - bz_1)}{\hat{d}_1} \quad i = 2 \dots N \\
 f_i &= (\Delta \hat{d}_i - \Delta d_1) \quad i = 2 \dots N
 \end{aligned} \tag{A.13}$$

Solucionando el sistema empleando la expresión A.8. se obtienen los incrementos a aplicar a la posición estimada del móvil mediante la ecuación A.9. Al igual que en el caso anterior este proceso se repite hasta que los incrementos sean inferiores a un umbral elegido por el diseñador.

Apéndice B

PDOP

El PDOP (*Position Dilution of Precision*) califica la geometría que tiene la disposición de las balizas con respecto a la posición del móvil y entrega un valor que permite conocer de forma general el grado de precisión vertical y horizontal que se tiene en la estimación de la posición del móvil.

Si sólo se tienen en cuenta los ejes de coordenadas X e Y , la medida es llamada HDOP (dilución de la precisión horizontal). Si sólo participa el eje Z la medida es llamada VDOP (dilución de la precisión vertical). Por lo tanto, el PDOP se puede obtener como:

$$PDOP = \sqrt{HDOP^2 + VDOP^2}$$

B.1. Cálculo del PDOP de forma teórica

El método de resolución empleando Gauss-Newton está muy extendido y estudiado, por lo cual existen diversas fórmulas para obtenerlo de forma teórica, [Nie97, SS02].

Retomando la matriz A de la ecuación A.6, el PDOP en un punto, puede obtenerse haciendo la raíz cuadrada de la traza de la matriz A^{-1}

$$A^{-1} = D = \begin{pmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{pmatrix} \quad (B.1)$$

$$PDOP = \sqrt{D_{11} + D_{22} + D_{33}}; \quad HDOP = \sqrt{D_{11} + D_{22}}; \quad VDOP = \sqrt{D_{33}}$$

En el caso de emplear el posicionamiento esférico, las derivadas que componen la matriz A hay que calcularlas empleando la expresión reflejada en la ecuación A.7, y en el caso hiperbólico empleando la ecuación A.13.

B.2. Cálculo del PDOP de forma experimental

Una alternativa de cálculo se basa en obtener de forma experimental un conjunto de medidas independientes de distancias entre la posición del móvil y la posición de las balizas. Calculando la posición $[x, y, z]$ del móvil para cada medida de distancias, de tal forma que se pueda calcular la varianza en el error de la posición $[x, y, z]$ del móvil para todo el conjunto de medidas hechas y poder así aplicar directamente la definición de PDOP.

Para simular el conjunto de medidas de distancia, se calculan las distancias entre el móvil y las N balizas, luego se reproduce el vector M veces, siendo M el número de medidas a simular. Posteriormente a esta matriz se le suma una matriz de igual dimensión con realizaciones de un ruido gaussiano de media cero y desviación típica σ_o .

$$\begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{pmatrix} \rightarrow \begin{pmatrix} d_1 & d_1 & \cdots & d_1 \\ d_2 & d_2 & \cdots & d_2 \\ \vdots & \vdots & \ddots & \vdots \\ d_N & d_N & \cdots & d_N \end{pmatrix}_{NxM} + \begin{pmatrix} \alpha_{11} & \alpha_{12} & \alpha_{1M} \\ \alpha_{21} & \alpha_{22} & \alpha_{2M} \\ \vdots & \vdots & \vdots \\ \alpha_{N1} & \alpha_{N2} & \alpha_{NM} \end{pmatrix}_{NxM} \quad (B.2)$$

Luego se calcula la posición del móvil para cada columna de la matriz resultante, con esto se obtiene una matriz de dimensiones $3 \times M$, que contiene las coordenadas $[x, y, z]$ de la posición del móvil para cada medida.

$$\begin{pmatrix} x_1 & x_2 & \cdots & x_M \\ y_1 & y_2 & \cdots & y_M \\ z_1 & z_2 & \cdots & z_M \end{pmatrix} \begin{matrix} \rightarrow \sigma_x \\ \rightarrow \sigma_y \\ \rightarrow \sigma_z \end{matrix} \quad (B.3)$$

Calculando la desviación típica en el error de posición para cada coordenada $[\sigma_x, \sigma_y, \sigma_z]$ es posible aplicar la definición de PDOP:

$$PDOP = \frac{\text{Error de posición}}{\text{Error de medida}} = \frac{\sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}}{\sigma_o} \quad (B.4)$$

Función de minimización de MATLAB

fmincon

La función *fmincon* de MATLAB permite encontrar el mínimo de una función, dentro de unas restricciones establecidas. Para ello emplea el algoritmo “Trust-Region-Reflective Optimization”, este algoritmo iterativo se acerca a la solución final mediante el método del gradiente conjugado preacondicionado (PCG) [CL92, CL93].

En este caso la función a minimizar es la explicada en la sección 4.3:

$$h = \text{Trace} (\mathbf{A}^T \mathbf{A})^{-1} \quad (\text{C.1})$$

Las restricciones para cada punto corresponden al tamaño del área de cobertura tanto la coordenada X, como para la coordenada Y. En este caso no se incluye la coordenada Z ya que corresponde a la altura donde se realizarán las medidas.

Por último la función de Matlab necesita una posición inicial de los puntos que se desean encontrar. En este caso siempre se elige que la posición inicial de los puntos sea el centro del área de cobertura, de esa forma se asegura que no estén cerca de los límites del área de trabajo.

Una vez ejecutada la función, devuelve la posición donde se deben ubicar los puntos de test desconocidos para obtener los mejores resultados en la calibración.

Extended abstract and conclusions

(English version)

D.1. Introduction

D.1.1. Motivation

The Indoor Localization Systems (ILS) are intended to get the position of a mobile target at a specific instant inside a building. These systems are gaining importance due to the great number and interest of their applications: in [Obd09, BK09, TK10, C.02] can be found some examples of them. Due to all this considerations, leading companies (Mitsubishi, Cisco, IBM, Microsoft, Motorola. . .) are investing in localization technologies. In fact, according to a market study developed by IDTechex [IDT09] in the last year, companies that offer location-based services have tripled in size and the localization market is estimated to grow from \$ 153 million in 2009 to 2.6 billion dollars in 2018.

One of the most frequent solution is to place several beacons in the building in order to emulate the behaviour of the GPS. ILS's based on WiFi use the access points inside the building; the ones with GSM use the base stations; if the ILS is based on Ultrasounds or Infrared some beacons should be placed in the environment.

In order to make the localization, in these systems normally it is necessary to know the actual position of the beacons; this is usually done manually by measuring their location within a reference system that is a costly and time consuming task. This fact make these systems little portables and adaptable.

In this PhD the problem of calibrating a local position system (LPS) is tackled, that is: to obtain the position of the beacons without measuring them in order to carry out the localization of mobile objects. Moreover the global calibration of several independent LPS is also tackled, in this case, such LPS's do not share their coverage area but they will share the same reference origin.

It should be noticed that although the methods proposed have been developed to improve the properties of an ultrasonic system, it can be ported to any other technology that

obtains the position of a mobile object by using time of flight (TOA) or time of flight differences (DTOA) informations.

D.1.2. Background

The indoor location is a hot topic among the research community. It is still unresolved with enough precision or resolution. There are many techniques for finding the position of a mobile object; according to [MND06] the indoor localization systems can be divided into the following categories:

- **Tokenless systems:** The mobile object does not carry any additional device to perform the localization.
- **Token systems:** The mobile object carry a special device to perform the localization.
- **Distributed systems:** In these systems a large number of sensor are placed in the enviroment and each node calculates its relative position from its neighboring nodes.

One of the most common solution used to performe the localization inside a building is to emulate the GPS. Several beacons, acting as satellites, are installed inside the area where the localization of the mobile objects is required. To implement these systems two different approaches can be used:

- **Installing a dedicated infrastructure to perform the localization:** In this case several devices are placed on the building to perform the localization. This is the case of systems based on infrared [WHFG92], ultrasounds [WHFG92, PCB00], Ultrawideband [SG05] and RFID [HBS08, HVBW01].
- **Usin devices already available:** These systems use devices previously installed in the building to perform the localization, for instance WiFi access points [BP00], Bluethooth networks [FMP05], or mobile phone infrastructure [OVLdL05, VDH⁺07].

Figure D.1 shows the accuracy of each of the main technologies available and their current development status. Figure D.2 as well shows the dominance of each of the technologies in typical positioning systems.

Once the positioning system has been deployed, different mathematical techniques can be used to locate the mobile object. According to [SJP⁺09] these techniques can be classified as:

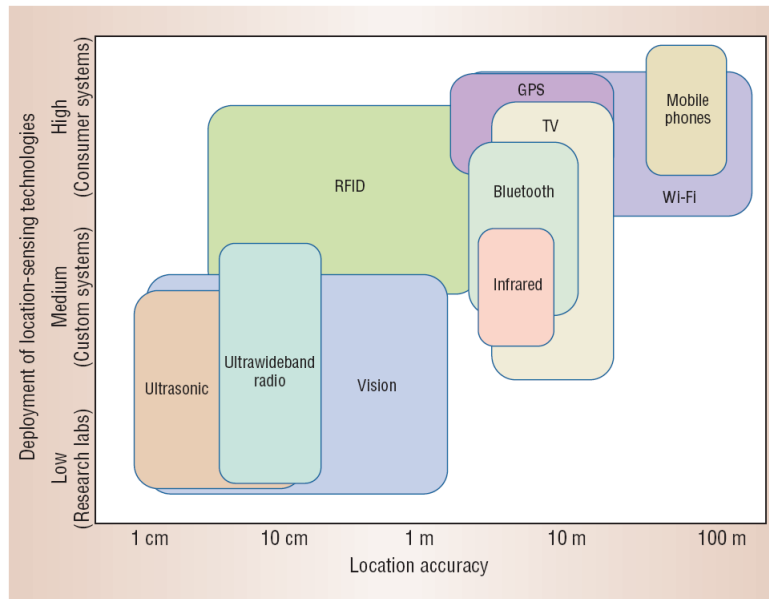


Figure D.1: Current status of different indoor location technologies: development and accuracy. Image extracted from [HSK04]

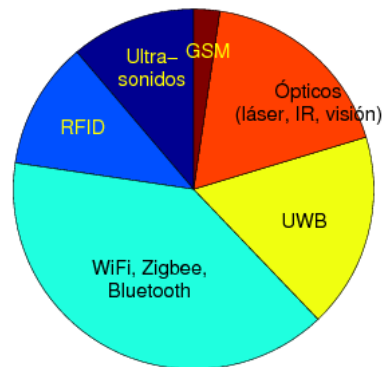


Figure D.2: Prevalence of based indoor technology. Image extracted from [HSK04]

- **Geometrical techniques:** These methods can be used when it is possible to measure distances, differences in distances or angles between the beacons and the moving object. Even in this simple situation, the system of equations can be nonlinear and normally this system is oversized so it is not possible to solve it in a closed form. Some examples of these methods can be found in [NMH98, Caf00, NG97].
- **Minimizing a cost function:** Generally, these methods consist of finding the minimum of a cost function, normally using Gauss-Newton or Levenberg-Marquardt methods. The main drawback of it is that the convergence of the algorithm depends strongly on their initialization. Some examples of these methods are found in [CS98, MLG⁺08, NPJ08].

- **Fingerprinting methods:** These methods are used mainly in LPS systems that measures the strength of the received signal (RSS). They usually need two stages: a *calibration stage*, where measures are taken through a grid of positions inside the coverage area and the signal strength received from base stations is stored; the *localization stage* itself, that can be done only after the calibration stage is finished. In this stage the mobile object measures the signal strength received from the base stations and compares it with the stored data base; after a matching process an hypothesis about actual localization is made. [FL08, Yim08, BB05].
- **Bayesian methods:** In these methods the position of the moving object at time t is modeled as a function of probability based on the measurements obtained previously. There are many implementations of these filters: Kalman Filter is one of the classical methods and is optimal when the system distribution is Gaussian. Nowadays interest is focused on probabilistic methods such as the Particle Filters [KSJ⁺07, MSG⁺05].

In general, in all those techniques it is mandatory to know the position of the beacons in order to find the position of the mobile object. Therefore it is necessary to measure or calculate the position between each beacon and the reference system, and store this position for further computations. This fact makes such systems not very portable and adaptable as the calibration process usually involves a lot of time and cost. So, nowadays new methods that allows to perform this process in a more automatic way are under research.

In [MF99, NNHM04] several methods based on inverse positioning are proposed. Basically, they take some measurements from several known test points, then the beacons positions are obtained using the trilateration algorithm, but assuming that the test points are the beacons and the actual LPS beacons are considered as mobile objects. The drawback of these techniques is that it is still needed to manually measure the position of all the test points.

In [DM03] a calibration method for spherical trilateration based on the Levenberg-Marquardt algorithm is presented. The algorithm assumes an initial position of the beacons and the test points and apply the minimization algorithm of Levenberg-Marquardt to minimize the error between the distance obtained with the estimated beacons and the measured distances. As the search space can be very large, the algorithm is repeated 100 times with random values and a fitting function is used in order to refine the results. Later in [DMC⁺05] this algorithm is adapted to hyperbolic trilateration.

D.1.3. The aims of this thesis

The main objective of this thesis is to obtain a calibration method to get the beacon positions of the local positioning system with low error and low computational cost. This main objective has a series of partial ones:

- *Obtaining a new hyperbolic trilateration algorithm*

Currently the direct resolution of the distance equations using Gauss-Newton is one of the most extended positioning methods because its accuracy [YTBJ08]. One of its drawbacks is that it is an iterative method, so its computational cost can be high if the system requires high precision. Also this algorithm has high computational cost if the system uses hyperbolic trilateration. So in the first stage of this thesis a new more efficient localization algorithm is seeking, that is, to have the same accuracy as the method of direct solution by Gauss-Newton but with a lower computational cost.

- *Proposal of a autocalibration of a single LPS*

One of the most common ways to obtain the positions of the beacons in a LPS is to apply inverse positioning. Some measures to the beacons are taken from some test points with known positions and the localization algorithm is used to obtain the position of the beacons. This thesis proposes new algorithms that allow the calibration of LPS without knowing the location of all measurement test points. So, it is possible to have a fixed and small device to make the measures from the known test points, and then perform more measures on different parts of the coverage area, whose locations are unknown to the system.

Also it is analyzed if, knowing the system structure, it is possible to calculate the best areas to place the unknown test points in order to minimize the calibration errors.

- *Global calibration of several independent LPS's*

With the autocalibration method proposed, several LPS's can be calibrated locally, that is, each LPS has its own reference system. The ultimate objective of this thesis is to obtain an algorithm to calibrate a set of several LPS's in a certain environment all together in a common and global reference system. In order to achieve this objective it will be proposed a method that merges the information from the odometer of a mobile robot with the information provided by the LPS.

- *Experimental validation*

Finally it will be carried out a series of simulations and experimental tests on a real ultrasonic LPS to verify the properties and performances of the different methods proposed in the thesis.

D.2. Localization algorithm based on the Cayley-Menger geometric properties

In this section a new localization algorithm based on the Cayley-Menger mathematical operator [Som58] is proposed. The geometric properties of this operator allows to find the localization of a mobile node in a faster and equally accurate way than the iterative methods based on the direct solution of the distance equations [YTBJ08].

This chapter is divided as follows: in the first section the Cayley-Menger bideterminant is defined and some of its geometrical properties are analyzed; then a spherical positioning using this operator is explained; in the next section the implementation of a new hyperbolic localization algorithm using this operator is proposed; finally some results using this algorithm are presented.

D.2.1. Cayley-Menger bideterminant

According to [Som58] the Cayley-Menger bideterminant for two sequence of 'n' points $[\mathbf{p}_1, \dots, \mathbf{p}_n]$ and $[\mathbf{q}_1, \dots, \mathbf{q}_n]$ can be defined as:

$$D(\mathbf{p}_1, \dots, \mathbf{p}_n; \mathbf{q}_1, \dots, \mathbf{q}_n) = 2 \left(\frac{-1}{2} \right)^n \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & D(\mathbf{p}_1, \mathbf{q}_1) & D(\mathbf{p}_1, \mathbf{q}_2) & \cdots & D(\mathbf{p}_1, \mathbf{q}_n) \\ 1 & D(\mathbf{p}_2, \mathbf{q}_1) & D(\mathbf{p}_2, \mathbf{q}_2) & \cdots & D(\mathbf{p}_2, \mathbf{q}_n) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & D(\mathbf{p}_n, \mathbf{q}_1) & D(\mathbf{p}_n, \mathbf{q}_2) & \cdots & D(\mathbf{p}_n, \mathbf{q}_n) \end{vmatrix} \quad (\text{D.1})$$

Where $D(\mathbf{p}_i, \mathbf{q}_j)$ is the squared of the Euclidian distance between the point \mathbf{p}_i and the point \mathbf{q}_j .

If the two sequences of points are the same, the operator $D(\mathbf{p}_1, \dots, \mathbf{p}_n; \mathbf{p}_1, \dots, \mathbf{p}_n)$ can be abbreviated by $D(\mathbf{p}_1, \dots, \mathbf{p}_n)$. In this case the Cayley-Menger operator is named as Cayley-Menger determinant.

The geometric properties of the Cayley-Menger operator can be found in [TORC03].

D.2.2. Spherical positioning

In this section the spherical trilateration algorithm proposed in [TR05] is explained. Derived from this algorithm several new localization algorithms will be obtained adapted to the hyperbolic positioning.

Giving three fixed points in the space, which localizations from a reference origin are known $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$, the localization problem is to find the coordinates of another point, \mathbf{m} ,

by measuring the distances between the fixed points and the point \mathbf{m} (l_1, l_2, l_3). A graphic scheme of this problem is shown in figure D.3

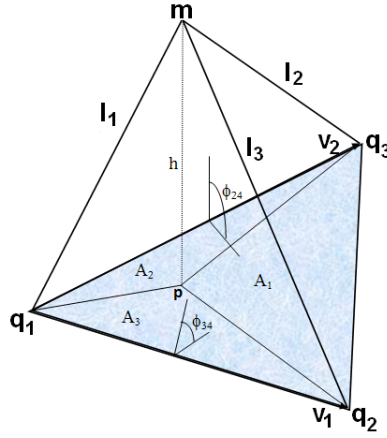


Figure D.3: Localization problem. Calculate the coordinates of the point ' \mathbf{m} ' using the measured distance l_1, l_2, l_3 to the points $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ respectively.

The first step to obtain the position of \mathbf{m} is to calculate the coordinates of its orthogonal projection, \mathbf{p} , to the base of the triangle created by the points $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$. In order to obtain the coordinates of \mathbf{p} , the baricentric coordinates are used [MLBD02]. This coordinates, (widely used in graphic design) allow to obtain the position of one point inside a triangle knowing the coordinates of the triangle vertex.

So using the baricentric coordinates and the properties of the Cayley-Menger operator the position of \mathbf{p} can be expressed as [TR05]:

$$\mathbf{p} = \mathbf{q}_1 - \frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_3, \mathbf{m})}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \mathbf{v}_1 + \frac{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{m})}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} \mathbf{v}_2 \begin{cases} \mathbf{v}_1 = \mathbf{q}_2 - \mathbf{q}_1 \\ \mathbf{v}_2 = \mathbf{q}_3 - \mathbf{q}_1 \end{cases} \quad (\text{D.2})$$

Once the position of \mathbf{p} is obtained following [TR05], the position of \mathbf{m} can be obtained as:

$$\mathbf{m} = \mathbf{p} \pm \frac{h}{\|\mathbf{v}_1 \times \mathbf{v}_2\|} (\mathbf{v}_1 \times \mathbf{v}_2) \quad (\text{D.3})$$

Using again the properties of the Cayley-Menger operator, the position of \mathbf{m} is calculated as:

$$\begin{aligned} \mathbf{m} = & \mathbf{q}_1 + \frac{1}{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)} (-D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_3, \mathbf{m}) \cdot \mathbf{v}_1 \\ & + D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3; \mathbf{q}_1, \mathbf{q}_2, \mathbf{m}) \cdot \mathbf{v}_2 \pm \left(\sqrt{D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{m})} \right) \cdot \mathbf{v}_3) \end{aligned} \quad (\text{D.4})$$

From this expression some conclusions can be obtained. First if the three fixed points are in line, $D(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ is zero so a singularity appears and it is impossible to find the position of \mathbf{m} . This also indicates that the larger the area between the three fixed points the lower the error in the location of the point \mathbf{m} for the same noise level in the measurements. This method obtains in a straightforward way the position of the point \mathbf{m} so it is not necessary to use iterative methods to get the solution.

D.2.3. Hyperbolic trilateration

One of the problems of the spherical trilateration is that, it is necessary a synchronization in the measures between the beacons and the mobile object. However in the hyperbolic trilateration is not necessary to know the emission moments of the beacons; this is because the position of the mobile object is obtained using the time difference between the reception of the signal from a reference beacon and the other ones. In this section, different hyperbolic localization algorithms based on the Cayley-Menger operator are presented.

The first step is to adapt the spherical equation (D.4) to the hyperbolic trilateration, that is, obtaining an expression in which the position of the mobile node were only function of the distance to the reference beacon and the difference in distance between the signal reception in each of the beacons with the reference one (figure D.4)

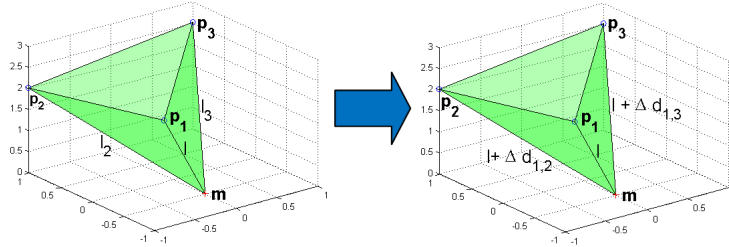


Figure D.4: From spherical to Hyperbolic trilateration being \mathbf{p}_1 the reference beacon.

Assuming that the reference beacon is \mathbf{p}_1 , let's call l the distance between this beacon and the mobile node, and $\Delta d(\mathbf{p}_j, \mathbf{p}_1)$ the difference of distance between the reception of the signal of the reference beacon and the reception of the signal of the beacon j . The position of \mathbf{m} can be expressed as:

$$\mathbf{m} = \mathbf{p}_1 + \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} [(b_1 \cdot l + c_1) \mathbf{v}_1 + (b_2 \cdot l + c_2) \mathbf{v}_2 \pm \left(\sqrt{a_3 \cdot l^2 + b_3 \cdot l + c_3} \right) \cdot \mathbf{v}_3] \quad (\text{D.5})$$

Where:

$$\begin{aligned}
 b_1 &= \frac{\Delta d(\mathbf{p}_3, \mathbf{p}_1)}{2} [D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_3)] - D(\mathbf{p}_1, \mathbf{p}_3) \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) \\
 b_2 &= \frac{\Delta d(\mathbf{p}_2, \mathbf{p}_1)}{2} [D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] + D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1) \\
 c_1 &= \frac{1}{4} \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \cdot [D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2)] \\
 &\quad + \frac{1}{4} \cdot D(\mathbf{p}_1, \mathbf{p}_3) \cdot [D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3) - 2\Delta d(\mathbf{p}_2, \mathbf{p}_1)^2] \\
 c_2 &= \frac{1}{4} \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot [D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] \\
 &\quad + \frac{1}{4} \cdot D(\mathbf{p}_1, \mathbf{p}_2) \cdot [D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_3) - 2\Delta d(\mathbf{p}_3, \mathbf{p}_1)^2] \\
 a_3 &= [D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_2, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1) \\
 &\quad - \frac{1}{4} \left\{ [D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2)]^2 + D(\mathbf{p}_2, \mathbf{p}_3)^2 - 2D(\mathbf{p}_1, \mathbf{p}_2) - 2D(\mathbf{p}_1, \mathbf{p}_3) \right\} \\
 &\quad - \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \tag{D.6} \\
 b_3 &= \frac{1}{4} \cdot \left\{ [D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_3) \right. \\
 &\quad + [D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_3) \\
 &\quad + [D(\mathbf{p}_1, \mathbf{p}_3) + D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2)] \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \cdot D(\mathbf{p}_1, \mathbf{p}_2) \left. \right\} \\
 &\quad - \frac{1}{4} \cdot \left\{ D(\mathbf{p}_1, \mathbf{p}_3) \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^4 + D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^4 \right. \\
 &\quad + [D(\mathbf{p}_{12}, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_2)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2 \\
 &\quad \left. + D(\mathbf{p}_1, \mathbf{p}_2) \cdot D(\mathbf{p}_1, \mathbf{p}_3) \cdot D(\mathbf{p}_2, \mathbf{p}_3) \right\} \\
 c_3 &= -D(\mathbf{p}_1, \mathbf{p}_3) \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) - D(\mathbf{p}_1, \mathbf{p}_2) \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1)^3 \\
 &\quad + \frac{1}{2} \cdot \{ [D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_2, \mathbf{p}_3) - D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_2, \mathbf{p}_1) \cdot D(\mathbf{p}_1, \mathbf{p}_3) \\
 &\quad + [-D(\mathbf{p}_1, \mathbf{p}_2) + D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_3)] \cdot \Delta d(\mathbf{p}_3, \mathbf{p}_1) \cdot D(\mathbf{p}_1, \mathbf{p}_2) \} \\
 &\quad + \frac{1}{2} \cdot \left\{ [\Delta d(\mathbf{p}_2, \mathbf{p}_1)^2 \Delta d(\mathbf{p}_3, \mathbf{p}_1) + \Delta d(\mathbf{p}_2, \mathbf{p}_1) \Delta d(\mathbf{p}_3, \mathbf{p}_1)^2] \right. \\
 &\quad \left. + [D(\mathbf{p}_1, \mathbf{p}_2) - D(\mathbf{p}_2, \mathbf{p}_3) + D(\mathbf{p}_1, \mathbf{p}_3)] \right\}
 \end{aligned}$$

As it can be observed in the equation D.5 the position of the point \mathbf{m} only depends on the distance to the reference beacon $\mathbf{m} = f(l)$, because the constants $b_1, c_1, b_2, c_2, a_3, b_3, c_3$ are known as the only depend on the measurements and the distances between the beacons. The process of the adaption of equations is explained in [RUG+11].

The proposed algorithms obtain the distance to the reference beacon, l , but it is necessary that the LPS has at least 4 beacons in order to create several triplets of beacons and thus have a system of equations. In the figure D.5 it can be shown that the position

of $\mathbf{m} = (m_x^{(1)}, m_y^{(1)}, m_z^{(1)})$ using the triplet $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ must be the same that the position of $\mathbf{m} = (m_x^{(2)}, m_y^{(2)}, m_z^{(2)})$ using the triplet $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)$. In the following paragraphs the superscript “(1)” will refer to the parameters obtained with the first triplet, and the superscript “(2)” to those obtained with the second triplet.

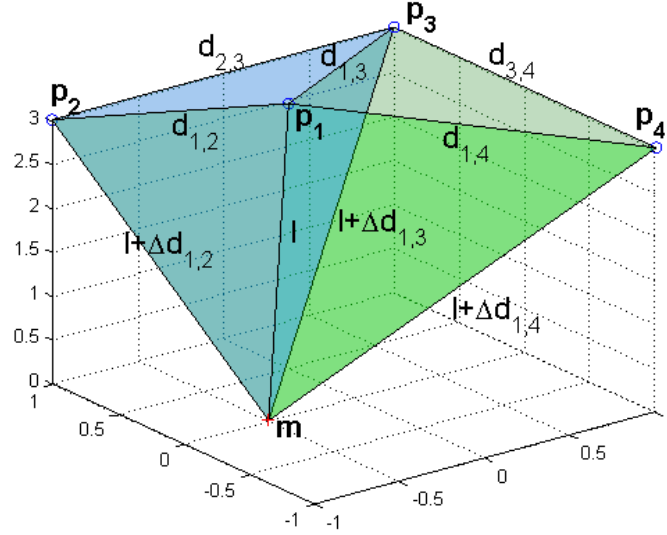


Figure D.5: Hyperbolic trilateration localization scheme.

So equating the position of \mathbf{m} obtained with the two triplets it is obtained the following nonlinear system of equations:

$$\mathbf{m}^{(1)} = \mathbf{m}^{(2)} \Rightarrow \begin{pmatrix} m_x^{(1)} \\ m_y^{(1)} \\ m_z^{(1)} \end{pmatrix} = \begin{pmatrix} m_x^{(2)} \\ m_y^{(2)} \\ m_z^{(2)} \end{pmatrix}$$

Once the value of l is calculated, the mobile position is obtained solving the equation D.5. The following sections introduce three methods to solve the location problem from different perspectives.

D.2.3.1. Method 1: LZC (Localization at constant Z).

If all beacons are at the same height the nonlinear system of equations can be transformed in a linear one, and it is possible to find the value of l in a straightforward way:

$$\left\{ \begin{array}{l} \frac{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[\left(b_1^{(1)} \cdot l + c_1^{(1)} \right) \cdot v_{x1}^{(1)} + \left(b_2^{(1)} \cdot l + c_2^{(1)} \right) \cdot v_{x2}^{(1)} \right] = \left[\left(b_1^{(2)} \cdot l + c_1^{(2)} \right) \cdot v_{x1}^{(2)} \right. \\ \left. + \left(b_2^{(2)} \cdot l + c_2^{(2)} \right) \cdot v_{x2}^{(2)} \right] \\ \frac{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left[\left(b_1^{(1)} \cdot l + c_1^{(1)} \right) \cdot v_{y1}^{(1)} + \left(b_2^{(1)} \cdot l + c_2^{(1)} \right) \cdot v_{y2}^{(1)} \right] = \left[\left(b_1^{(2)} \cdot l + c_1^{(2)} \right) \cdot v_{y1}^{(2)} \right. \\ \left. + \left(b_2^{(2)} \cdot l + c_2^{(2)} \right) \cdot v_{y2}^{(2)} \right] \end{array} \right. \quad (\text{D.7})$$

The whole process and some examples under simulation of this method can be found in [RUG+09a]

D.2.3.2. Method 2: Localization with Rotated Triplets

This method proposes a rotation of the reference origin (figure D.6) so at least three beacons are at the same plane of constant Z. This transforms the nonlinear system in a quadratic one, thus it is also possible to find the value of l in a straightforward way:

$$\left\{ \begin{array}{l} \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left(\left(b_1^{(1)} l + c_1^{(1)} \right) \mathbf{v}_{x1}^{(1R)} + \left(b_2^{(1)} l + c_2^{(1)} \right) \mathbf{v}_{x2}^{(1R)} \right) = \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)} \left[\left(b_1^{(2)} l + c_1^{(2)} \right) \mathbf{v}_{x1}^{(2R)} \right. \\ \left. + \left(b_2^{(2)} l + c_2^{(2)} \right) \mathbf{v}_{x2}^{(2R)} \pm \left(\sqrt{a_3^{(2)} l^2 + b_3^{(2)} l + c_3^{(2)}} \right) \cdot \mathbf{v}_{x3}^{(2R)} \right] \\ \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)} \left(\left(b_2^{(1)} l + c_2^{(1)} \right) \mathbf{v}_{y2}^{(1R)} \right) = \frac{1}{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4)} \left[\left(b_1^{(2)} l + c_1^{(2)} \right) \mathbf{v}_{y1}^{(2R)} \right. \\ \left. + \left(b_2^{(2)} l + c_2^{(2)} \right) \mathbf{v}_{y2}^{(2R)} \pm \left(\sqrt{a_3^{(2)} l^2 + b_3^{(2)} l + c_3^{(2)}} \right) \cdot \mathbf{v}_{y3}^{(2R)} \right] \end{array} \right. \quad (\text{D.8})$$

The superscript R indicates that the point has been rotated. The whole process and some examples under simulation of this method can be found in [RUG+11]

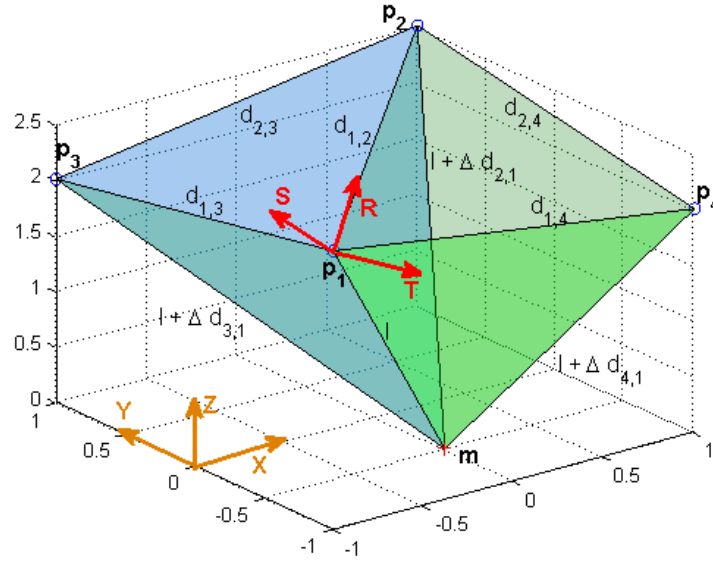


Figure D.6: Localization scheme rotating the reference origin.

D.2.3.3. Method 3: LTI Localization using an iterative process

In this method the system is solved using a nonlinear algorithm, specifically the Gauss-Newton algorithm. This algorithm tries to find the value of l through an iterative process, starting from an initial value, \hat{l}_0 , the algorithm find a value of \hat{l}_k that makes all the positions obtained by all the triplets equals (figure D.7).

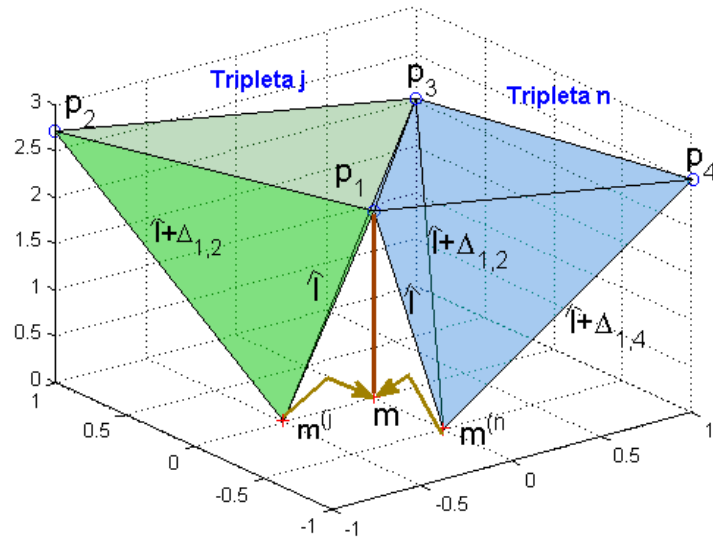


Figure D.7: Localization scheme for the non-linear method.

All information about this method and some results under simulation are detailed in [URG+11]

D.2.4. Actual test

To verify the properties of the proposed algorithms several test are performed in an actual ultrasonic LPS [VUM⁺07, PUH⁺07]. The figure D.8 shows the LPS used on tests: all beacons are at the same height and a conical reflector is used to increase the coverage area.

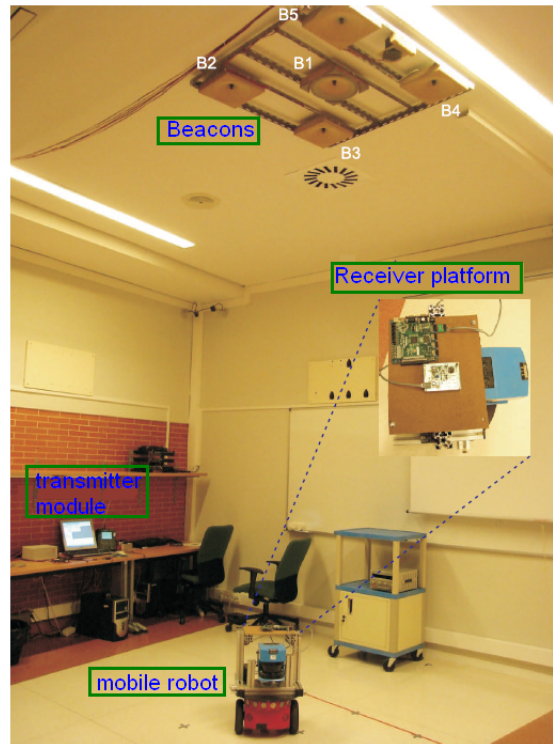


Figure D.8: Ultrasonic LPS.

In these tests, a path inside the coverage area is performed and the results obtained are compared with the localization results using the direct resolution of the distance equation using Gauss-Newton, which is chosen because this algorithm is widely used in the LPS systems [YTBJ08].

The figure D.9 presents the localization results using the method LZC and the direct resolution of the distance equations using Gauss-Newton. The mean difference in the X and Y coordinates between the two methods is 11.4 mm. and 10.8 mm respectively. this difference is a bit higher because the real height of the beacons is not constant due to the conical reflector. The main advantage of the method LZC is that is about 5 times in faster getting the positions.

In the figure D.10 it can be observed the same path but this time using the method LTI. In this case the different between the algorithms is 1.8 mm in the X axis and 1.98 mm in the Y axis so it can be concluded that both algorithms have the same accuracy. The proposed algorithm has the advantage that is able to filter some outliers in the measurements, and is about 4 times faster.

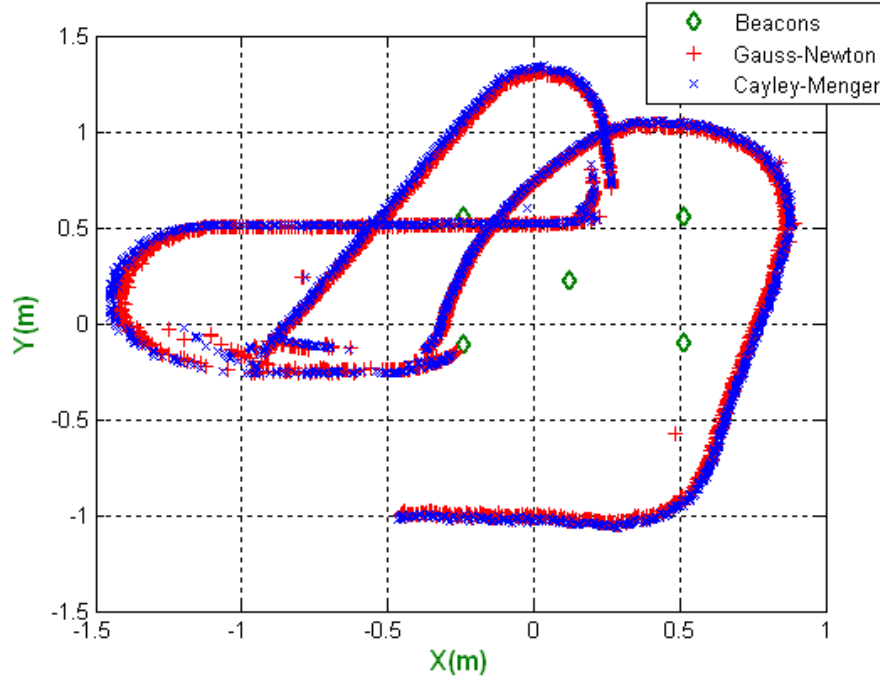


Figure D.9: Comparison between the direct resolution of the equations using Gauss-Newton and the first proposed method based on Cayley-Menger operator, making a path inside the coverage area.

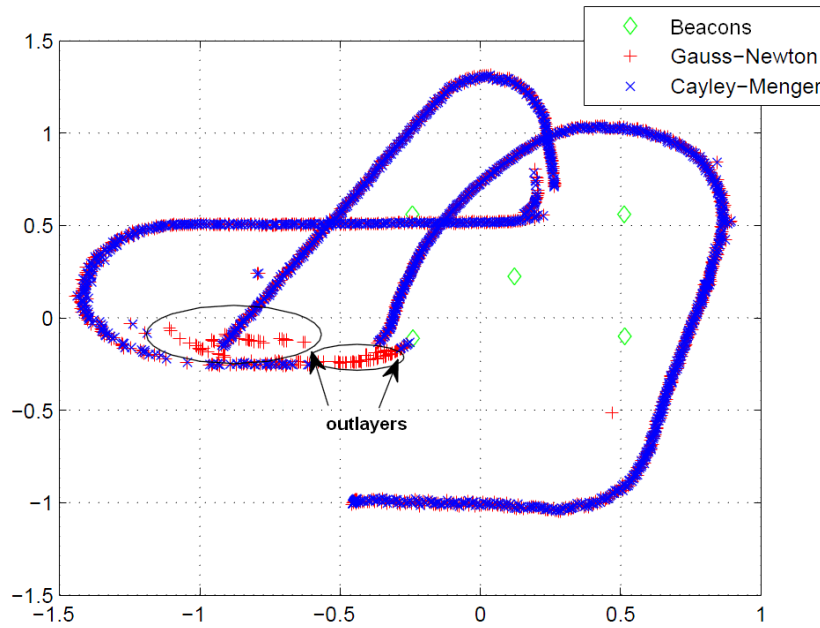


Figure D.10: Comparison between the direct resolution of the equations using Gauss-Newton and the third proposed method based on Cayley-Menger operator, making a path inside the coverage area.

More tests can be found in [URG+11], [RUG+09a] and [RUG+11]. The method 2 is not tested with the arrangement in figure D.8 because it can only be used when the beacons do not have the same height.

D.3. Calibration algorithm for a beacon based LPS system

In this chapter an algorithm for calibrating a LPS is proposed. Two versions of the algorithm are presented: one for the spherical case and another for the hyperbolic case. Next the method to find the optimal areas to perform the calibration is analyzed. Finally some actual results are shown.

In order to calibrate a LPS system, several measures to the beacons are normally taken from several test points inside the coverage area. Once known the position of these points it is possible to find the position of the beacons just using the localization algorithm in a inverse way [MF99]. These methods have the disadvantage that they still need to manually measure the position of all test points.

In the proposed algorithm it is not necessary to know the localization of all test points but only the localization of three of them must be known. The first one defines the origin of the reference system, this fact eliminates ambiguities in the solution due to translations of the coordinate system. The other two points eliminate ambiguities in the solution due to rotations of the reference system.

The initial data for the proposed method are:

- A vector containing: the initial estimated values assigned to each of the variables of the algorithm, **pai**, in this case; the coordinates of the N beacons $(\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi})$, with $i = 1 \dots N$; and the localization of the M unknown test points (\hat{x}_j, \hat{y}_j) with $j = 1 \dots M$. The Z component of each test point is assumed to be known.
- The distances between the test points and the beacons. In the spherical case this is the absolute distance between them (r_i) . In the hyperbolic case this is the distance from each beacon to the test point minus the distance from the test point to the reference beacon (Δr_i) .
- The localization of the three known test points $\mathbf{X}_a, \mathbf{X}_b, \mathbf{X}_c$

This algorithm uses the iterative resolution method of Gauss-Newton to bring the random initial estimates of the beacon and the unknown test points localization, **pai**, to their real localization.

D.3.1. Spherical version

The distance between the beacon i with coordinates (b_{xi}, b_{yi}, b_{zi}) and the test point \mathbf{m} with coordinates (x_m, y_m, z_m) is:

$$r_{im} = \sqrt{(b_{xi} - x_m)^2 + (b_{yi} - y_m)^2 + (b_{zi} - z_m)^2} \quad (\text{D.9})$$

So, the distance between the estimated beacon i with coordinates $(\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi})$ and the estimated test point \mathbf{m} with coordinates $(\hat{x}_m, \hat{y}_m, \hat{z}_m)$ is:

$$\hat{r}_{im} = \sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} \quad (\text{D.10})$$

The algorithm uses the iterative method of Gauss-Newton [Fle87] to minimize the sum of squared errors between the measured distances and the ones obtained from the estimated values of the system using the following expression:

$$F = \sum_{i=1}^N \sum_{m=4}^M (\hat{r}_{im} - r_{im})^2 = \sum_{i=1}^N \sum_{m=4}^M [f_{im}]^2 \quad (\text{D.11})$$

Where

$$f_{im} = \sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} - r_{im} \quad (\text{D.12})$$

Differentiating the expression D.12 over all the unknowns of the system and organizing it as $\mathbf{A} \cdot \Delta \mathbf{X} = \mathbf{B}$, it is obtained:

$$\begin{aligned} \mathbf{A} &= \begin{pmatrix} \frac{\partial f_{11}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial f_{11}}{\partial \hat{b}_{zN}} & \frac{\partial f_{11}}{\partial \hat{x}_4} & \dots & \frac{\partial f_{11}}{\partial \hat{y}_M} \\ \frac{\partial f_{12}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial f_{12}}{\partial \hat{b}_{zN}} & \frac{\partial f_{12}}{\partial \hat{x}_4} & \dots & \frac{\partial f_{12}}{\partial \hat{y}_M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{NM}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial f_{NM}}{\partial \hat{b}_{zN}} & \frac{\partial f_{NM}}{\partial \hat{x}_4} & \dots & \frac{\partial f_{NM}}{\partial \hat{y}_M} \end{pmatrix} \\ \Delta \mathbf{X} &= [\hat{b}_{x1}, \dots, \hat{b}_{x1}, \hat{x}_4, \dots, \hat{y}_m]^T \\ \mathbf{B} &= [f_{11}, \dots, f_{NM}]^T \end{aligned} \quad (\text{D.13})$$

Where:

$$\begin{aligned}
\frac{\partial f_{im}}{\partial b_{xi}} &= \frac{\hat{b}_{xi} - \hat{x}_m}{\hat{r}_{im}}; & \frac{\partial f_{im}}{\partial x_m} &= \frac{\hat{x}_m - \hat{b}_{xi}}{\hat{r}_{im}} \\
\frac{\partial f_{im}}{\partial b_{yi}} &= \frac{\hat{b}_{yi} - \hat{y}_m}{\hat{r}_{im}}; & \frac{\partial f_{im}}{\partial y_m} &= \frac{\hat{y}_m - \hat{b}_{yi}}{\hat{r}_{im}} \\
\frac{\partial f_{im}}{\partial b_{zi}} &= \frac{\hat{b}_{zi} - \hat{z}_m}{\hat{r}_{im}}; & \frac{\partial f_{im}}{\partial z_m} &= \frac{\hat{z}_m - \hat{b}_{zi}}{\hat{r}_{im}}
\end{aligned}
\quad \text{with } \begin{cases} i = 1 \dots N \\ m = 4 \dots M \end{cases} \quad (\text{D.14})$$

Solving the system using least squares:

$$\Delta \mathbf{X} = (\mathbf{A}^T \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{B} \quad (\text{D.15})$$

Once the position differences $\Delta \mathbf{X}$ are obtained , the vector of the estimated values is updated following the expression:

$$\mathbf{pai}_{k+1} = \mathbf{pai}_k - \Delta \mathbf{X}_k$$

This process is repeated until the $\Delta \mathbf{X}$ are lower than a threshold.

The whole algorithm and some results under simulation can be found in [RUJ+08]

D.3.2. Hyperbolic version

In this case instead of the absolute distance between the test points and the beacons, what it is measured is the distance between the reception of the signal from the reference beacon and the rest of them (Δr_i). It is assumed, without loss of generality, that the first beacon is the reference one.

The increment of distance between the beacon i with coordinates (b_{xi}, b_{yi}, b_{zi}) and the test point \mathbf{m} with coordinates (x_m, y_m, z_m) can be defined as:

$$\begin{aligned}
\Delta r_{im} = r_{im} - r_{1m} &= \sqrt{(b_{xi} - x_m)^2 + (b_{yi} - y_m)^2 + (b_{zi} - z_m)^2} \\
&\quad - \sqrt{(b_{x1} - x_m)^2 + (b_{y1} - y_m)^2 + (b_{z1} - z_m)^2}
\end{aligned} \quad (\text{D.16})$$

Also the distance between the estimated beacon i with coordinates $(\hat{b}_{xi}, \hat{b}_{yi}, \hat{b}_{zi})$ and the estimated test point \mathbf{m} with coordinates $(\hat{x}_m, \hat{y}_m, \hat{z}_m)$ can be defined as:

$$\begin{aligned} \Delta \hat{r}_{im} = \hat{r}_{im} - \hat{r}_{1m} = & \sqrt{(\hat{b}_{xi} - \hat{x}_m)^2 + (\hat{b}_{yi} - \hat{y}_m)^2 + (\hat{b}_{zi} - \hat{z}_m)^2} \\ & - \sqrt{(\hat{b}_{x1} - \hat{x}_m)^2 + (\hat{b}_{y1} - \hat{y}_m)^2 + (\hat{b}_{z1} - \hat{z}_m)^2} \end{aligned} \quad (D.17)$$

Using the same strategy as in the previous case, the squared error distance between the distances measured and the estimated ones is iteratively minimized using the following equation:

$$G = \sum_{i=2}^N \sum_{m=4}^M (\Delta \hat{r}_{im} - \Delta r_{im})^2 = \sum_{i=2}^N \sum_{m=4}^M [g_{im}]^2 \quad (D.18)$$

Differentiating D.18 like in the spherical case and grouping the system as $\mathbf{A} \cdot \Delta \mathbf{X} = \mathbf{B}$ it is obtained:

$$\begin{aligned} \mathbf{A} = & \begin{pmatrix} \frac{\partial g_{21}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial g_{21}}{\partial \hat{b}_{zN}} & \frac{\partial g_{21}}{\partial \hat{x}_4} & \dots & \frac{\partial g_{21}}{\partial \hat{y}_M} \\ \frac{\partial g_{22}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial g_{22}}{\partial \hat{b}_{zN}} & \frac{\partial g_{22}}{\partial \hat{x}_4} & \dots & \frac{\partial g_{22}}{\partial \hat{y}_M} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_{NM}}{\partial \hat{b}_{x1}} & \dots & \frac{\partial g_{NM}}{\partial \hat{b}_{zN}} & \frac{\partial g_{NM}}{\partial \hat{x}_4} & \dots & \frac{\partial g_{NM}}{\partial \hat{y}_M} \end{pmatrix} \\ \Delta \mathbf{X} = & [\hat{b}_{x1}, \dots, \hat{b}_{x1}, \hat{x}_4, \dots, \hat{y}_m]^T \\ \mathbf{B} = & [f_{11}, \dots, f_{NM}]^T \end{aligned} \quad (D.19)$$

Where:

$$\begin{aligned} \frac{\partial f_{im}}{\partial b_{xi}} &= \frac{\hat{b}_{xi} - \hat{a}_m}{\hat{r}_{im}}; & \frac{\partial f_{im}}{\partial b_{x1}} &= \frac{\hat{x}_m - \hat{b}_{x1}}{\hat{r}_{1m}}; \\ \frac{\partial f_{im}}{\partial b_{yi}} &= \frac{\hat{b}_{yi} - \hat{a}_m}{\hat{r}_{im}}; & \frac{\partial f_{im}}{\partial b_{y1}} &= \frac{\hat{y}_m - \hat{b}_{y1}}{\hat{r}_{1m}}; \\ \frac{\partial f_{im}}{\partial b_{zi}} &= \frac{\hat{b}_{zi} - \hat{a}_m}{\hat{r}_{im}}; & \frac{\partial f_{im}}{\partial b_{z1}} &= \frac{\hat{z}_m - \hat{b}_{z1}}{\hat{r}_{1m}}; \\ \frac{\partial f_{im}}{\partial x_m} &= \frac{\hat{x}_m - \hat{b}_{xi}}{\hat{r}_{im}} - \frac{\hat{x}_m - \hat{b}_{xi}}{\hat{r}_{im}} & \text{with } \begin{cases} i = 2 \dots N \\ m = 4 \dots M \end{cases} \\ \frac{\partial f_{im}}{\partial y_m} &= \frac{\hat{y}_m - \hat{b}_{yi}}{\hat{r}_{im}} - \frac{\hat{y}_m - \hat{b}_{yi}}{\hat{r}_{im}} \\ \frac{\partial f_{im}}{\partial z_m} &= \frac{\hat{z}_m - \hat{b}_{zi}}{\hat{r}_{im}} - \frac{\hat{z}_m - \hat{b}_{zi}}{\hat{r}_{im}} \end{aligned} \quad (D.20)$$

Solving the system using least squares it is possible to find $\Delta \mathbf{X}$, and then update the vector of the estimated values like in the spherical case.

The whole algorithm and some results under simulation can be found in [RUJ+08].

D.3.3. Optimal localization of the test points

As it was described in the previous section, the test points can be placed anywhere inside the coverage area, but depending on their location the measurement errors may affect most to the final solution of the system. For instance, which one is the better option: to place all of them under the beacons?, place them in the limits of the coverage area? or spread them into the whole coverage area?

In [RUV+08] it is presented a method to find the better areas to place the measurement points. At the end, it is demonstrated that to find these areas is necessary to minimize the function:

$$h = \text{Trace}(\mathbf{A}^T \mathbf{A})^{-1} \quad (\text{D.21})$$

The value of the matrix \mathbf{A} can be found in [RUV+08]

D.3.4. Actual Results

In this section some actual tests are presented. These tests have been performed in the LPS described in the section D.2.4. To verify the accuracy of the proposed algorithms, a manual calibration of the LPS was performed, and the resulting beacon coordinates are shown in the table D.1.

Table D.1: Beacons localization after the manual calibration

(m)	Beacon 1	Beacon 2	Beacon 3	Beacon 4	Beacon 5
Coordinate X	0.122 m.	0.513 m.	0.515 m.	-0.24 m.	-0.242 m.
Coordinate Y	0.224 m.	0.56 m.	-0.1 m.	-0.11 m.	0.56 m.
Coordinate Z	3.5 m.	3.5 m.	3.5 m.	3.5 m.	3.5 m.

For all tests the position of the known test points is shown in the Table D.2.

Table D.2: Coordinates [x, y, z] of the known test points.

Point 1	Point 2	Point 3
[0, 0, 0.03] m.	[0.5, 0, 0.03] m.	[0, 0.5, 0.03] m.

Table D.3: Position of the 8 unknown test points for the spherical calibration test, all units are in meters.

(m)	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7	Point 8
Coord. X	0.0	-1.5	2.0	-1.5	0.5	1.5	2.0	-1.5
Coord. Y	-1.5	-1.5	0.5	1.5	1.5	1.5	-1.5	0
Coord. Z	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03

The first test is performed using spherical trilateration and taking measurements from eight unknown test points, whose actual positions are shown in the table D.3. In the table D.4 the coordinates obtained after the 3D calibration are shown.

Table D.4: Beacons localization after the 3D spherical calibration with 8 unknown test points.

(m)	Beacon 1	Beacon 2	Beacon 3	Beacon 4	Beacon 5
Coordinate X	0.121	0.504	0.542	-0.234	-0.242
Coordinate Y	0.212	0.561	-0.109	-0.119	0.534
Coordinate Z	3.487	3.495	3.488	3.489	3.496

Figure D.11 shows the errors in each of the components (x, y, z) in the position of the calibrated beacons. It can be observed that most of the coordinates' error is less than 1.5 cm, and in this case the mean error is 2.04 cm.

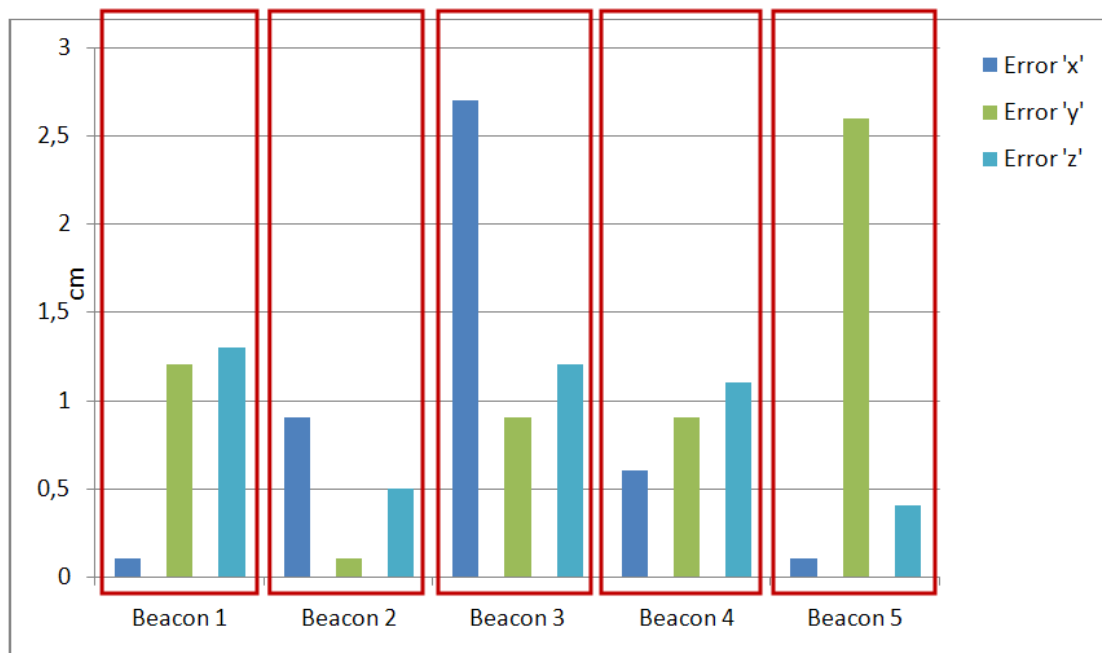


Figure D.11: Error in centimeters between the beacons coordinates obtained using the 3D spherical calibration algorithm proposed and their real coordinates.

In case of using hyperbolic trilateration, the system PDOP is very high, between 50 and 85 (figure D.12 a), so even with a low time of flight (TOF) measurement error, it has

not been possible to make a 3D calibration thereof even placing the unknown test points in the optimal areas.

Analyzing more in detail the PDOP, that is, broken it down into their horizontal (figure D.12 b) and vertical components (figure D.12 c), it appears that the higher errors are due to VDOP, as it varies between 50 and 70, while the HDOP has lower values (between 10 and 40). So if it is not possible to make a 3D calibration, it is more feasible to perform a 2D calibration, because in this case the vertical component is known, and the system is not affected by the VDOP.

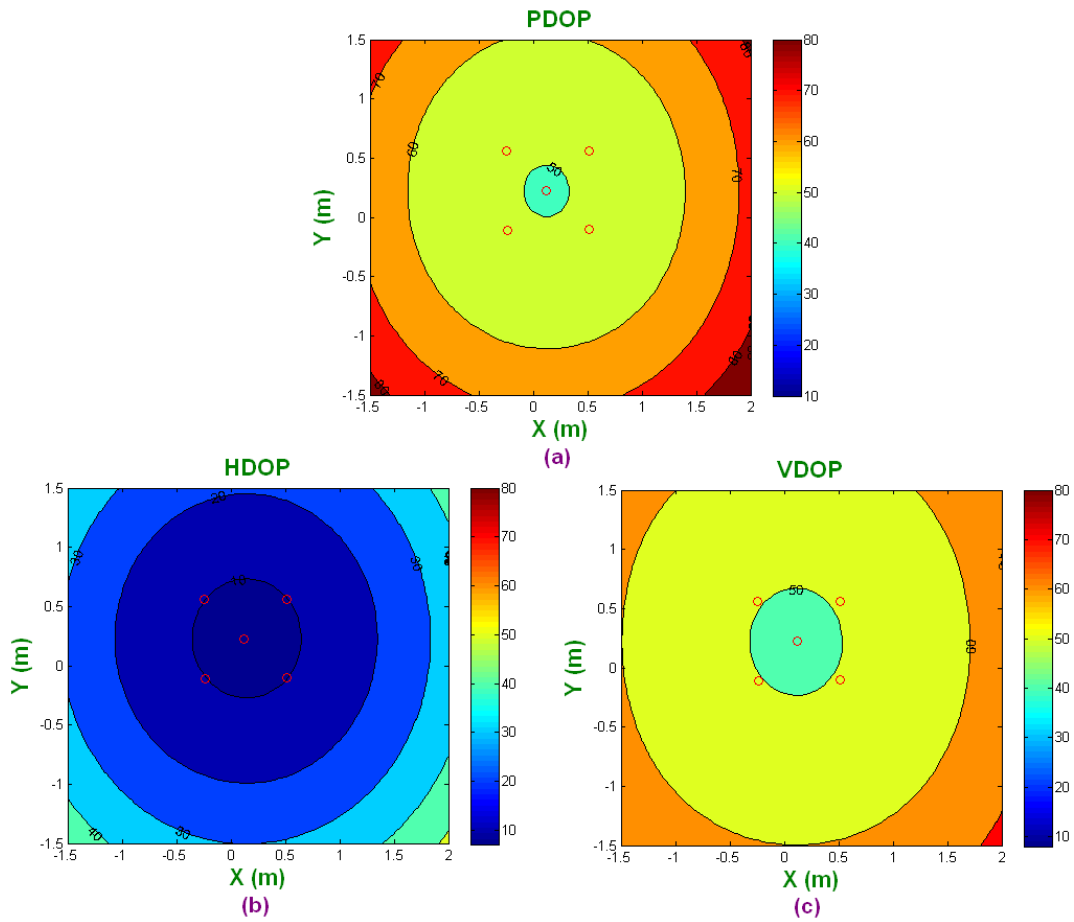


Figure D.12: LPS PDOP when hyperbolic trilateration is used inside the coverage area. (a) Hyperbolic PDOP. (b) Hyperbolic HDOP. (c) Hyperbolic VDOP.

In this experiment, using the algorithm of the optimal areas, the unknown test points were placed in the coordinates indicated in the table D.5. The position of the known test points is the same as in the previous case.

Table D.5: Position of the 8 unknown test points for the hyperbolic calibration test, all units are in meters.

(m)	Point 1	Point 2	Point 3	Point 4	Point 5	Point 6	Point 7	Point 8
Coord. X	0	-1.5	-1.5	-1.5	0.5	1.5	2.0	-1.5
Coord. Y	-1.5	-1.5	0.0	1.5	1.5	1.5	-1.5	0
Coord. Z	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03

In the table D.6 the coordinates of the calibrated beacons are shown. In this case the mean error is 3.21 cm. Moreover in the figure D.13 the error between the real beacons and the calibrated ones are shown, it can be observer than even using a 2D calibration the errors are higher than in the case of using spherical trilateration. The errors are around 1.5 cm in most of the cases.

Table D.6: Beacons localization after the 2D hyperbolic calibration with 8 unknown test points. The coordinate Z is known.

(m)	Beacon 1	Beacon 2	Beacon 3	Beacon 4	Beacon 5
Coordinate X	0.135	0.497	0.551	-0.205	-0.229
Coordinate Y	0.207	0.546	-0.145	-0.122	0.541
Coordinate Z	3.5	3.5	3.5	3.5	3.5

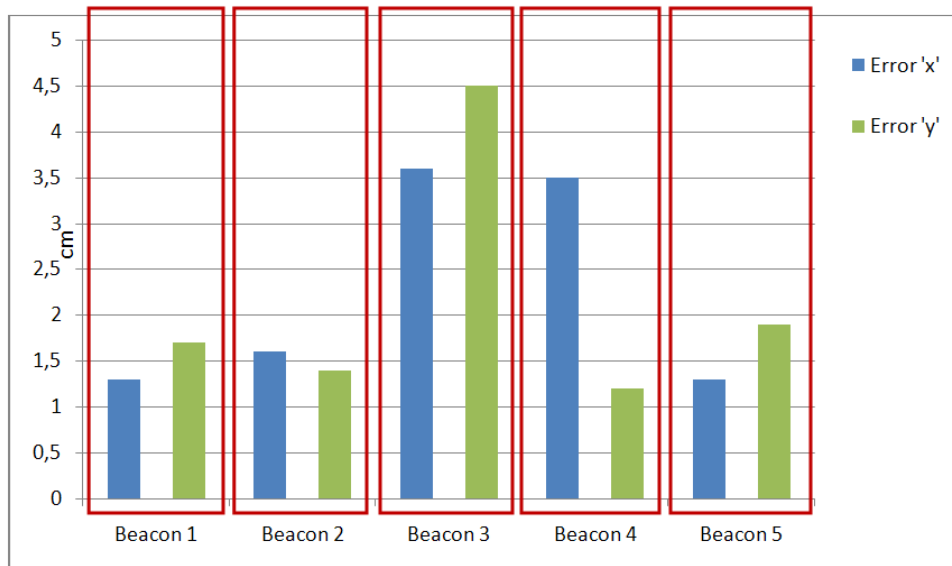


Figure D.13: Error in centimeters between the beacons coordinates obtained using the 2D hyperbolic calibration algorithm proposed and their real coordinates.

More real test and some paths performed using the calibrated beacons can be found in [RUJ+08] and [RUV+08].

D.4. Global calibration of several ultrasonic LPSs using a mobile robot

In the previous chapter, a calibration algorithm for a single LPS has been proposed. Present chapter is focused on a global calibration method for several independent LPS. In this context, global calibration means that all systems will share the same origin of reference.

A mobile robot is used in order to perform the calibration of all LPS; the proposed algorithm will merge the data from the robot odometer (Linear and Angular speed) with the information provided by the LPS using an $H-\infty$ filter. Through an iterative process the algorithm calculates both the robot's position and the beacons' localization.

D.4.1. Calibration of a single LPS

Once the robot reaches the coverage area of an LPS a calibration process is carried out. This process is divided into 3 stages (figure D.14):

- **Precalibration:** With the first set of measurements getting from both the odometer and the LPS system, a first estimation of the beacons position is obtained using an inverse positioning algorithm.
- **Calibration:** The robot moves inside the coverage area to improve the beacons' position obtained in the previous step. In this stage the algorithm merges the data obtained by the odometer and the LPS using an $H-\infty$ filter.
- **Navigation:** Once the calibration process is finished the position of the beacons are introduced to the system and the $H-\infty$ filter is used for robot navigation.

D.4.1.1. Precalibration

This stage is needed because to perform the calibration stage it is required an initial position of the beacons. In this stage a small path is performed inside the coverage area and an inverse positioning algorithm is used to locate the beacons; that is: the robot positions during the path are stored as if they were beacons. With the distances measured from these points to the beacons, a localization algorithm is used to locate each of the beacons (figure D.15). In this proposal it is used the localization algorithm based on the properties of the Cayley-Menger bideterminants, explained in the chapter D.2.

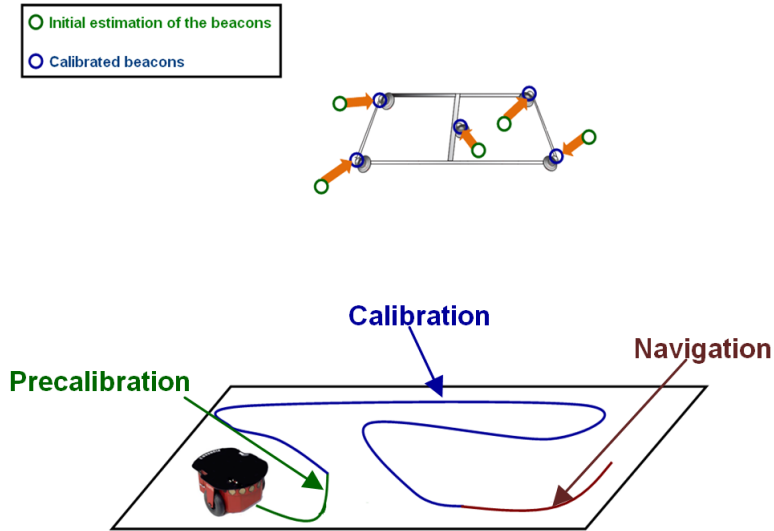


Figure D.14: Phases of the self-calibration algorithm. Precalibration, Calibration and Navigation.

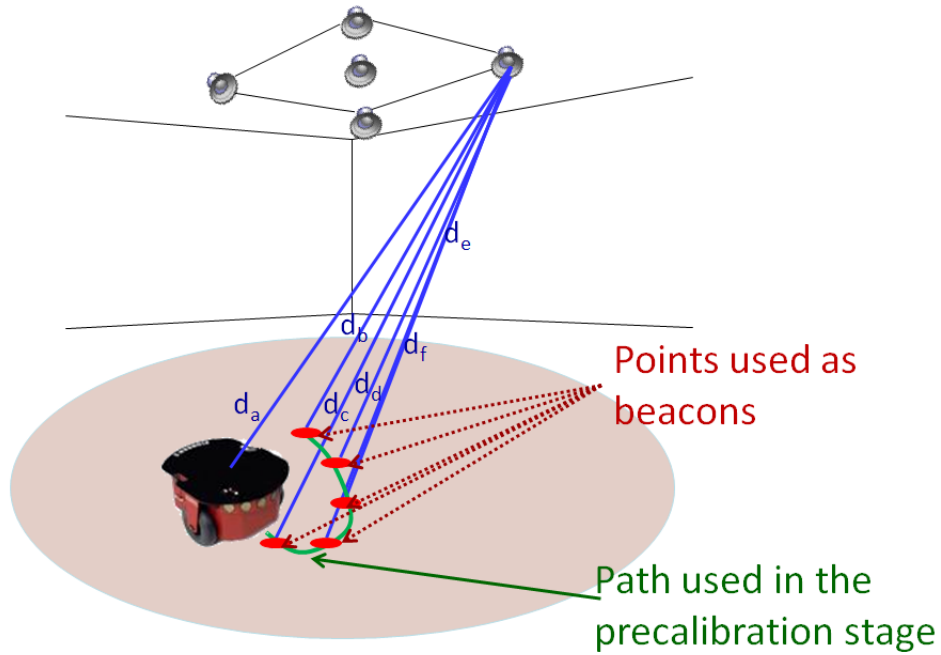


Figure D.15: Example of how to obtain the position of one beacon using the distances measured between the mobile robot and the beacon on several points of the trajectory.

D.4.1.2. Calibration

One of the fusing techniques more used for mobile robots navigation is the Kalman Filter. In this work instead of the Kalman Filter an H_∞ Filter is proposed as method of data fusion. The main advantage of H_∞ Filter in present application is that it minimizes the maximum estimation error. In [ST92] is proved that the H_∞ Filter is less sensitive to the parametr variations than the Kalman Filter; the advantage is that H_∞ Filter can be used though a fully reliable system model is not known.

Following [SD97] the H- ∞ filter equations can be expressed in the following two stages:

- Prediction stage:

$$\hat{\mathbf{q}}_{k,odo} = f(\hat{\mathbf{q}}_{k-1}, odometer)$$

- Update stage

$$\begin{aligned} \mathbf{L}_k &= (\mathbf{I} - \gamma \cdot \mathbf{Q}_k \cdot \mathbf{P}_k + \mathbf{C}_k^T \cdot \mathbf{V}^{-1} \cdot \mathbf{C}_k \cdot \mathbf{P}_k)^{-1} \\ \mathbf{K}_k &= \mathbf{A}_k \cdot \mathbf{P}_k \cdot \mathbf{L}_k \cdot \mathbf{C}_k^T \cdot \mathbf{V}^{-1} \\ \hat{\mathbf{q}}_k &= \hat{\mathbf{q}}_{k,odo} + \mathbf{K}_k (\mathbf{d}_k - \hat{\mathbf{d}}_k) \\ \mathbf{P}_{k+1} &= \mathbf{A}_k \cdot \mathbf{P}_k \cdot \mathbf{L}_k \cdot \mathbf{A}_k \cdot \mathbf{W} \end{aligned} \tag{D.22}$$

Where the meaning of the different elements in the equations are:

- $\hat{\mathbf{q}}_{k,odo}$ is the estimated vector that contains the robot position, the robot orientation and the beacons positions, using only the odometry data.
- $f(\hat{\mathbf{q}}_{k-1}, odometer)$ is the function that obtains the robot position at time k , using the position of the robot at time $k-1$ and the odometry measurements.
- $\hat{\mathbf{q}}_k$ contains the estimated variables, the localization and the orientation of the robot and the beacons localization after the update stage.
- \mathbf{K}_k is the filter gain.
- \mathbf{d}_k contains the measured distances between the robot and the beacons.
- $\hat{\mathbf{d}}_k$ are the estimated measurements, in this case the distance between the estimated position of the robot and the estimated position of the beacons.
- \mathbf{A}_k and \mathbf{C}_k are the Jacobian matrix of the system dynamics. In this case the robot and ultrasound dynamics.
- \mathbf{V} and \mathbf{W} are matrix related with the noise of both the robot odometry and the ultrasonic processes.
- \mathbf{I} is the identity matrix.
- \mathbf{P}_{k+1} is the filter covariance.
- $\gamma \cdot \mathbf{Q}$ is a parameter of the filter.

All these parameters for the case of the odometer and the LPS system can be found in [RUG+09b].

D.4.1.3. Navigation

This phase starts once the calibration stage is finished. In this stage the position of the beacons are no longer an unknown of the system, and the $H-\infty$ Filter only has to estimate the position and orientation of the mobile robot. In this situation the filter equations are the same, but with a reduced number of components in the state vector number of components, because the beacons localization is known.

The new equations can be found in [RUG+10a].

D.4.2. Extension to several LPS

Once the first LPS is calibrated, the objective is to calibrate another LPS system inside the building, but both of them sharing the same reference origin, so a global calibration of all systems is obtained (Figure D.16)

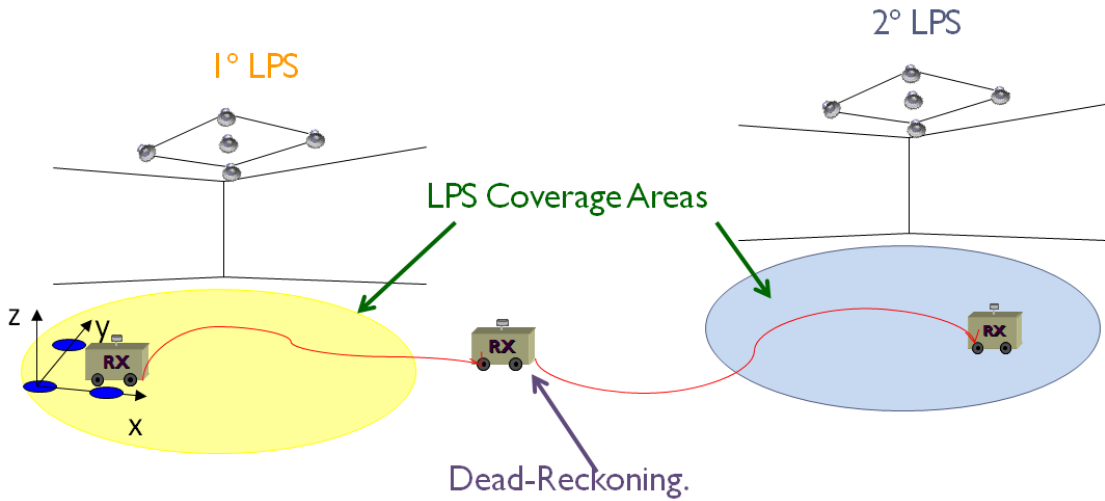


Figure D.16: Global System calibration of several LPS.

When the calibration of the first LPS is finished the robot is guided to the next LPS and when the new LPS is detected by the robot a new calibration process is started. In the path between one LPS to the other the robot only can use dead-reckoning. The errors in these areas cannot be corrected, so they affect to the calibration of the new LPS. When the calibration in the new LPS is performed the algorithm obtains a right structure of the beacons but shifted and rotated regarding the actual position.

The error in the calibration of the second LPS is because the bad estimation of the point P_2 , when the second calibration process starts (Figure D.17). Errors in the shift and the rotation of this point are transferred to the position of the beacons after the calibration. To minimize this effect, in [RUG+10a] is proposed a method to increase the accuracy in the estimation of this point by using several trips between the LPSs.

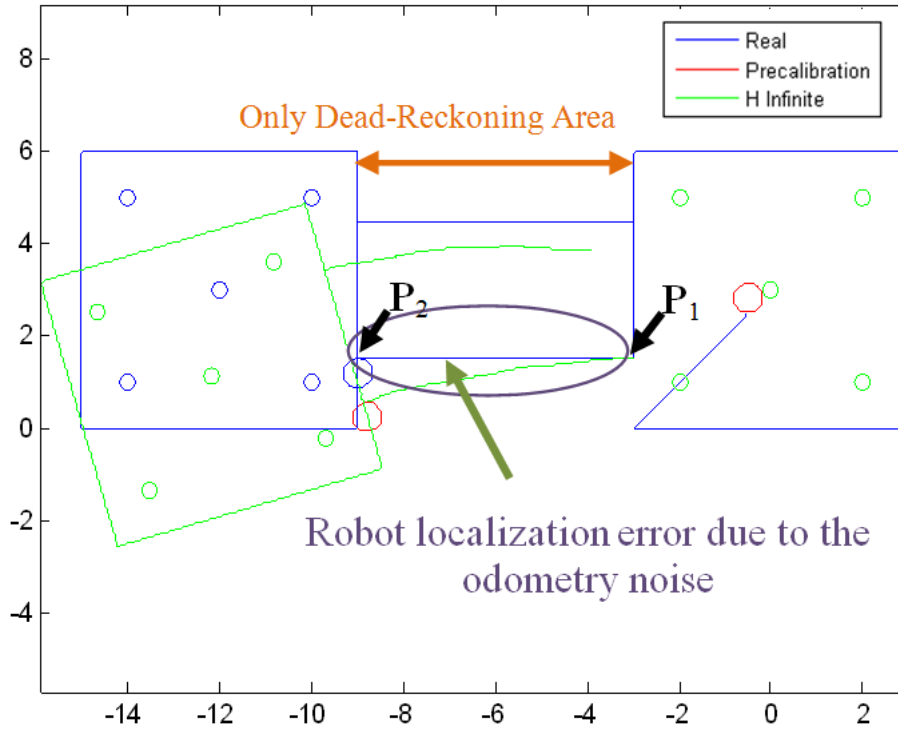


Figure D.17: Error in the beacons position estimation in the second LPS due to the noise in the robot odometry. Circles are the projection of the beacons on the ground.

D.4.3. Actual Results

This section presents the results obtained in the real ultrasonic LPS system. In a practical situation it is very difficult to get the real positions of the beacons from the origin of references. In the test the geometric structure of the real beacons is compared with the one obtained in the calibration process.

The figure D.18 shows the path used to perform the calibration process. The black circles represent the positions obtained from the odometer while the blue crosses represent the estimated positions obtained with the filter; the points used for the precalibration have been remarked.

The figure D.19 shows the results obtained comparing the distances between beacons. It can be observed that the beacon structure after the calibration process is much closer to the actual structure than the one obtained after the precalibration process. In this test the mean distance error is reduced by nearly half, from 2.50 cm to 1.35 cm.

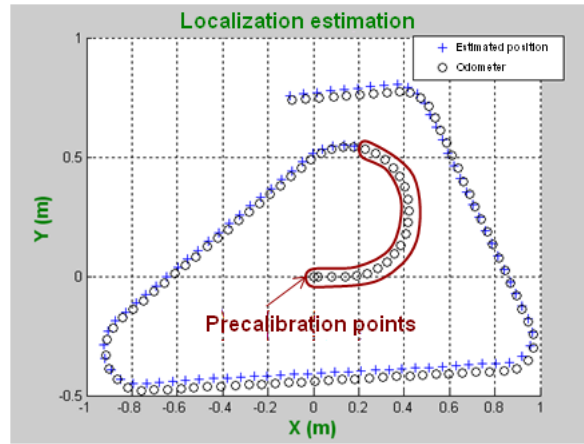


Figure D.18: Path performed with the mobile robot in order to test the proposed calibration algorithm. Also the points used for the precalibration stage are shown.

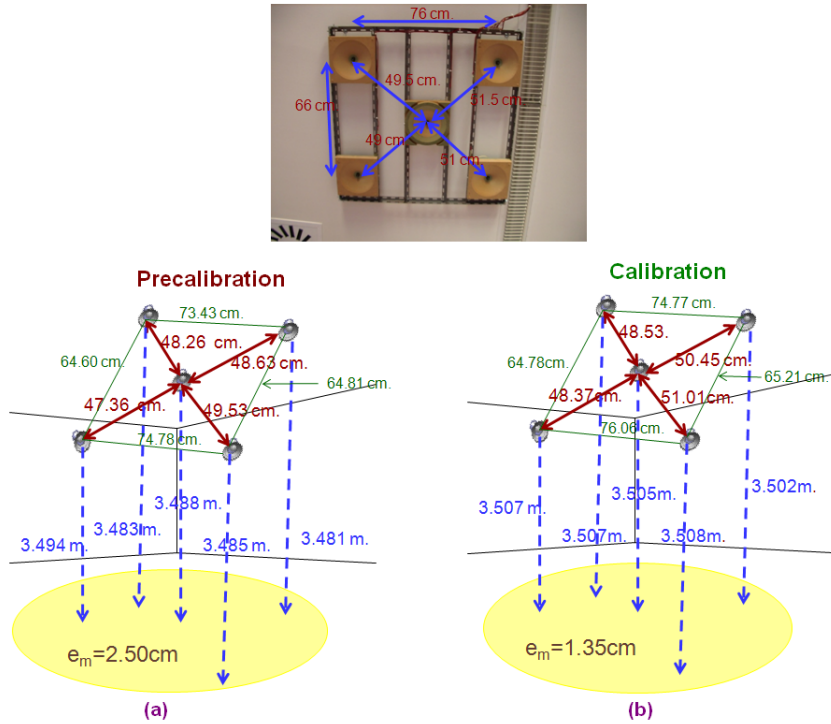


Figure D.19: Comparison between the beacons obtained in the precalibration and in the calibration stage. The height of the beacons is 3.5 m.

In the current stage of the research there are no real results involving two different LPS's because there is only one LPS available. Currently an additional LPS is under development. Simulated results of several LPS calibration can be found in [RUG+09b] and [RUG+10a].

D.5. Conclusions and future works

Conclusions

In this thesis the problem of the calibration of a local positioning system has been analyzed, that is finding a mechanism to obtain the actual localization of the system beacons, as this information is needed to calculate the position of a mobile node inside the coverage area. The main contributions of this thesis can be summarized as follows:

- *New hyperbolic trilateration algorithms.*

A new hyperbolic trilateration algorithm (derived from the spherical trilateration method in [TR05]) has been proposed. The spherical equations have been adapted to the hyperbolic trilateration and afterwards three new methods have been proposed to solve the new nonlinear equations. These methods have proved (in simulated and actual tests) to have the same performance than the traditional iterative methods [YTBJ08] but being more computationally efficient.

The first method (LZC) is only possible if all beacons are at the same height, this situation is quite common as the beacons are usually placed on the ceiling of the room. Under these circumstances the nonlinearities of the system equations are eliminated and the position of the mobile node can be found by solving a linear system.

The second method (LTR) transforms the system of equations in a quadratic system using a rotation of the system of equations in order to get at least three beacons at the same height. The problem of this method is that its precision is worse than the traditional ones.

The last proposed method (LTI) can be used in any circumstance and solve the nonlinear system of equations using the iterative method of Gauss-Newton.

- *Proposal of an autocalibration method for a single LPS*

A method to calibrate a single LPS system has been presented for the case of using spherical or hyperbolic trilateration. The proposed method does not need to know the actual coordinates of all the measuring test points to find the position of the beacons, it only need to know three of them. Also a method that allows to know the best area to place the measuring test points is proposed.

In the simulated and actual tests it has been proved that placing the measuring test points in the optimal areas minimize the errors in the calibration; also it has been proved that the results using spherical trilateration are better than the hyperbolic ones.

In the case of using hyperbolic trilateration, specially if the system PDOP is very high, in order to decrease the errors in the calibrations is better to perform a 2D calibration and measure manually the height of the beacons

■ *Design of a global calibration method for several LPS*

A global calibration method for multiple independent LPS's using a mobile robot have been presented. In a first stage the algorithm performs the calibration of the first LPS using a H- ∞ Filter. Once the calibration is finished the mobile robot moves to the next LPS and performs the calibration of the new system, using the reference system of the first LPS.

When the robot navigates between a calibrated LPS and a non-calibrated LPS, it can travel inside areas without coverage from any LPS. The errors introduced by the odometer in this areas can affect severely in the calibration of the new LPS's, so a mechanism to reduce the errors in these areas has been proposed.

In actual and simulated tests it has been checked, that the errors in the calibration process are around centimeters, and that the calibration algorithm is robust to faults in the odometry. It has also be checked that the proposed method reduce the influence of the odometer errors in the areas where the robot has no support from any LPS reducing the original errors, in the order of decimeters, to the order of centimeters.

Once the calibration of the system is finished, the use of the H- ∞ Filter has been tested as a fusing algorithm of the information coming from the odometry and the LPS. In the different tests it has been proved its best performance and behaviour against other fusing algorithms typically employed in mobile robot guidance like the Extended Kalman Filter.

Future works

This thesis deals with the calibration problem in the indoor LPS's. The results have shown that some improvements have been achieved. However, a lot of work is yet to be done in this field; specifically, some of the task to be performed are:

■ *Development and calibration of multiple ultrasonic LPS*

It was not possible to test in a real scenario with multiple LPS the proposed global calibration algorithm based on the H- ∞ Filter. Currently new ultrasonic LPS's are being developed with wide coverage área but without the conical reflectors in order to avoid the uncertainties in the positions of the emitters.

■ *Study of another fusing techniques*

In the global calibration algorithm an H- ∞ Filter fuses the information between the odometer and the LPS, it can be studied how affects to the calibration another fusing algorithms as the Particle Filters.

■ *Study the location of the known test points*

In the single calibration method it has been proposed an algorithm to find the optimal areas to locate the unknown test points. It can be studied how the position of the known test points affect to the errors in the calibration.

Bibliografía

- [ADB⁺04] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita. A line in the sand: A wireless sensor network for target detection, classification, and tracking. *Computer Networks (Elsevier)*, 46:605–634, 2004.
- [ATSW04] I. Ashokaraj, A. Tsourdos, P. Silson, and B. A. White. Sensor based robot localisation and navigation: using interval analysis and unscented Kalman filter. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2004)*, volume 1, pages 7–12, 2004.
- [BB05] Mauro Brunato and Roberto Battiti. Statistical learning theory for location fingerprinting in wireless LANs. *Comput. Netw. ISDN Syst.*, 47:825–845, April 2005.
- [BK09] T. Baldwin and L.T. Kuriakose. Cheap, accurate RFID tracking of museum visitors for personalized content delivery. In *Proc. Museums and the Web*, 2009.
- [Bol08] Philipp Bolliger. Redpin - adaptive, zero-configuration indoor localization through user collaboration. In *MELT '08: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 55–60, New York, NY, USA, 2008. ACM.
- [BP00] P. Bahl and V. N. Padmanabhan. Radar: an in-building RF-based user location and tracking system. In *Proc. IEEE Nineteenth Annual Joint Conf. of the IEEE Computer and Communications Societies INFOCOM 2000*, volume 2, pages 775–784, 2000.
- [BP03] T. Lyon G. Brignone, C. Connors and S. Pradhan. Smartlocus: An autonomous, self-assembling sensor network for indoor asset and systems management. *Technical Report HPL-2003-41, HP Labs*, 2003.

- [BW04] R. Bischoff and R. Wattenhofer. Analyzing connectivity-based multi-hop ad-hoc positioning. In *Proc. Second IEEE Annual Conf. Pervasive Computing and Communications PerCom 2004*, pages 165–174, 2004.
- [C.02] Pinhanez C. Creating ubiquitous interactive games using everywhere displays projectors. In *Proc. of the International Workshop on entertainment Computing (IWEC)*, 2002.
- [Caf00] Jr. Caffery, J. J. A new approach to the geometry of TOA location. In *Proc. 52nd Vehicular Technology Conf. IEEE VTS-Fall VTC 2000*, volume 4, pages 1943–1949, 2000.
- [CBAC09] Marc Ciurana, Francisco Barcelo-Arroyo, and Sebastiano Cugno. Tracking mobile targets indoors using WLAN and Time of Arrival. *Comput. Commun.*, 32(13-14):1552–1558, 2009.
- [CC92] F. Chenavier and J. L. Crowley. Position estimation for a mobile robot using vision and odometry. In *Proc. Conf. IEEE Int Robotics and Automation*, pages 2588–2593, 1992.
- [Che08] K. C. Cheok. Ultra-wideband (UWB) local positioning system (LPS) - an experimental study and analysis. In *Proc. World Automation Congress WAC 2008*, 2008.
- [CL92] Thomas F. Coleman and Yuying Li. On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds. Technical report, Ithaca, NY, USA, 1992.
- [CL93] Thomas F. Coleman and Yuying Li. An interior trust region approach for nonlinear minimization subject to bounds. Technical report, Ithaca, NY, USA, 1993.
- [Cor08] Panasonic Corporation. Omnidirectional back electret condenser microphone cartridge, Noviembre 2008.
- [CS98] Jr. Caffery, J. and G. L. Stuber. Subscriber location in CDMA cellular networks. 47(2):406–416, 1998.
- [DBS96] M. Deffenbaugh, J. G. Bellingham, and H. Schmidt. The relationship between spherical and hyperbolic positioning. In *Proc. MTS/IEEE. 'Prospects for the 21st Century' OCEANS '96*, volume 2, pages 590–595, 1996.
- [Dev08] Analog Devices. Ssm2166: Microphone preamplifier wirth variable compression noise gating, Noviembre 2008.

- [DM03] P. Duff and H. Muller. Autocalibration algorithm for ultrasonic location systems. In *Proc. Seventh IEEE Int Wearable Computers Symp*, pages 62–68, 2003.
- [DMC⁺05] Paul Duff, Michael McCarthy, Angus Clark, Henk Muller, Cliff Randell, Cliff R, Shahram Izadi, Andy Boucher, Andy Law, Richard Swinford, and Sarah Pennington. A new method for auto-calibrated object tracking. In *Top. Appl*, pages 123–140. Springer-Verlag, 2005.
- [eka11] www.ekahau.com. 2011.
- [FAM⁺05] Lei Fang, P. J. Antsaklis, L. A. Montestruque, M. B. McMickell, M. Lemmon, Yashan Sun, Hui Fang, I. Koutroulis, M. Haenggi, Min Xie, and Xiaojuan Xie. Design of a wireless assisted pedestrian dead reckoning system - the NavMote experience. *IEEE Transactions on Instrumentation and Measurement*, 54(6):2342–2358, 2005.
- [FL08] Shih-Hau Fang and Tsung-Nan Lin. Indoor location system based on discriminant-adaptive neural network in IEEE 802.11 environments. 19(11):1973–1978, 2008.
- [Fle87] Robert Fletcher. *Practical methods of optimization (2nd ed.)*. John Wiley & Sons, 1987.
- [FMP05] F. Forno, G. Malnati, and G. Portelli. Design and implementation of a bluetooth ad hoc network for indoor positioning. *IEE Proceedings - Software*, 152(5):223–228, 2005.
- [For91] K. Forsman. *Applications of Gröbner Bases to Nonlinear Systems*, 0 edition, 1991.
- [Foy76] W. H. Foy. Position-location solutions by Taylor-series estimation. (2):187–194, 1976.
- [GB07] S. A. Golden and S. S. Bateman. Sensor measurements for Wi-Fi location with emphasis on Time-of-Arrival ranging. *IEEE Transactions on Mobile Computing*, 6(10):1185–1198, 2007.
- [GK05] Craig Gotsman and Yehuda Koren. Distributed graph layout for sensor networks. *Journal of Graph Algorithms and Applications*, 9-(3):327–346, 2005.
- [HBS08] Yanchuan Huang, P. V. Brennan, and A. Seeds. Active RFID location system based on time-difference measurement using a linear FM chirp

- tag signal. In *Proc. IEEE 19th Int. Symp. Personal, Indoor and Mobile Radio Communications PIMRC 2008*, pages 1–5, 2008.
- [HHS⁺02] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. *Wirel. Netw.*, 8(2/3):187–197, 2002.
- [HSK04] M. Hazas, J. Scott, and J. Krumm. Location-aware computing comes of age. *Computer*, 37(2):95–97, 2004.
- [HVBW01] Jeffrey Hightower, Chris Vakili, Gaetano Borriello, and Roy Want. Design and calibration of the SpotON Ad-Hoc location sensing system. Technical report, 2001.
- [IDT09] IDTechex. Real time locating systems, 2009.
- [Inc08a] Measurement Specialist Inc. 40 khz omni-directional ultrasound transmitter, application specification document, Noviembre 2008.
- [Inc08b] Mobile Robots Inc. Pioneer 3-dx: The general purpose robot base, Noviembre 2008.
- [Kar06] I. H. Kartowisastro. Local positioning system of a mobile robot : A practical perspective. In *Proc. 9th Int. Conf. Control, Automation, Robotics and Vision ICARCV '06*, pages 1–6, 2006.
- [Kir02] E. Kiriy. *A Localization System for Autonomous Golf Course Mowers*. PhD thesis, Dept. of Electrical Engineering, McGill University, 2002.
- [KSJ⁺07] A. D. Koutsou, F. Seco, A. R. Jimenez, J. O. Roa, J. L. Ealo, C. Prieto, and J. Guevara. Preliminary localization results with an RFID based indoor guiding system. In *Proc. IEEE Int. Symp. Intelligent Signal Processing WISP 2007*, pages 1–6, 2007.
- [Kur04] Derek Kurth. Range-only robot localization and SLAM with radio. Master's thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 2004.
- [LHAR99] T. D. Larsen, K. L. Hansen, N. A. Andersen, and Ole Ravn. Design of Kalman filters for mobile robots; evaluation of the kinematic and odometric approach. In *Proc. IEEE Int Control Applications Conf*, volume 2, pages 1021–1026, 1999.

- [LMP⁺09] C. Losada, M. Mazo, S. Palazuelos, D. Pizarro, M. Marron, and F. Redondo. 3D motion segmentation and 3D localization of mobile robots using an array of static cameras and objective function minimization. In *Proc. IEEE Int. Symp. Intelligent Signal Processing WISP 2009*, pages 299–304, 2009.
- [LW07] Konrad Lorincz and Matt Welsh. MoteTrack: a robust, decentralized approach to RF-based location tracking. *Personal Ubiquitous Comput.*, 11(6):489–503, 2007.
- [Ma06] Min Zhu Yanmin Ni Lionel M. Ma, Jian Gao. Anchor-free localization with refinement in sensor networks. Technical report, HKUST Institutional Repository, 2006.
- [MF99] Ajay Mahajan and Fernando Figueroa. An automatic self-installation and calibration method for a 3D position sensing system using ultrasonics. *Robotics and Autonomous Systems*, 28(4):281 – 294, 1999.
- [ML09] A. R. Jiménez y F. Seco M. Laguna, J. Roa. Diversified local search for the optimal layout of beacons in an indoor positioning system. In *IIE Transactions vol. 41, no. 3, pp. 247-259*, 2009.
- [MLBD02] Mark Meyer, Haeyoung Lee, Alan Barr, and Mathieu Desbrun. Generalized barycentric coordinates on irregular polygons. *Journal of Graphics Tools*, 7:13–22, 2002.
- [MLC⁺06] Hyun Myung, Hyoung-Ki Lee, Kiwan Choi, Seokwon Bang, YeunBae Kim, and Sangryong Kim. Mobile robot localization using a gyroscope and constrained Kalman Filter. In *Proc. Int SICE-ICASE Joint Conf*, pages 2098–2103, 2006.
- [MLG⁺08] S. Mazuelas, F. A. Lago, D. Gonzalez, A. Bahillo, J. Blas, P. Fernandez, R. M. Lorenzo, and E. J. Abril. Dynamic estimation of optimum path loss model in a RSS positioning system. In *Proc. IEEE/ION Position, Location and Navigation Symp*, pages 679–684, 2008.
- [MMKP02] Randolph L. Moses, Olph L. Moses, Dushyanth Krishnamurthy, and Robert Patterson. A self-localization method for wireless sensor networks. *EURASIP Journal on Applied Signal Processing*, 4:348–358, 2002.
- [MND06] E. D. Manley, H. A. Nahas, and J. S. Deogun. Localization and tracking in sensor systems. In *Proc. IEEE Int Sensor Networks, Ubiquitous, and Trustworthy Computing Conf*, volume 2, pages 237–242, 2006.

- [MSG⁺05] M. Marron, M. A. Sotelo, J. C. Garcia, D. Fernandez, and D. Pizarro. "XPFCP": An extended Particle Filter for tracking multiple and dynamic objects in complex environments. In *Proc. IEEE Int. Symp. Industrial Electronics ISIE 2005*, volume 4, pages 1587–1592, 2005.
- [NF02] L. Naimark and E. Foxlin. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. In *Proc. Int. Symp. Mixed and Augmented Reality ISMAR 2002*, pages 27–36, 2002.
- [NG97] S. C. Nardone and M. L. Graham. A closed-form solution to bearings-only target motion analysis. 22(1):168–178, 1997.
- [Nie97] R. O. Nielsen. Relationship between dilution of precision for point positioning and for relative positioning with GPS. 33(1):333–338, 1997.
- [NKY08] Yusuke Nakazato, Masayuki Kanbara, and Naokazu Yokoya. Localization system for large indoor environments using invisible markers. In *VRST '08: Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, pages 295–296, New York, NY, USA, 2008. ACM.
- [NMH98] William Navidi, William S. Murphy, and Willy Hereman. Statistical methods in surveying by trilateration. *Computational Statistics & Data Analysis*, 27(2):209–227, April 1998.
- [NN03] D. Niculescu and Badri Nath. Ad hoc positioning system (APS) using AOA. In *Proc. INFOCOM 2003. Twenty-Second Annual Joint Conf. of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1734–1743, 2003.
- [NNHM04] A. Nishitani, Y. Nishida, T. Hori, and H. Mizoguchi. Portable ultrasonic 3D tag system based on a quick calibration method. In *Proc. IEEE Int Systems, Man and Cybernetics Conf*, volume 2, pages 1561–1568, 2004.
- [NPJ08] A. Narzullaev, Yongwan Park, and Hoyoul Jung. Accurate signal strength prediction based positioning for indoor WLAN systems. In *Proc. IEEE/ION Position, Location and Navigation Symp*, pages 685–688, 2008.
- [OA00] Robert J. Orr and Gregory D. Abowd. The smart floor: a mechanism for natural user identification and tracking. In *CHI '00: CHI '00 extended abstracts on Human factors in computing systems*, pages 275–276, New York, NY, USA, 2000. ACM.

- [Obd09] D. Obdrzalek. Small autonomous robot localization system. In *Proc. IEEE Student Conf. Research and Development (SCORED)*, pages 461–463, 2009.
- [OVLdL05] Veljo Otsason, Alex Varshavsky, Anthony LaMarca, and Eyal de Lara. Accurate GSM indoor localization. In *IN THE PROC. OF UBICOMP 2005*, pages 141–158, 2005.
- [PB07] V. Chook S. Chessa A. Gotta Y. Hu P. Baronti, P. Pillai. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. In *Computer Communications*, volume 30, pages 1655–1695, May 2007.
- [PCB00] Nissanka Bodhi Priyantha, Anit Chakraborty, and Hari Balakrishnan. The Cricket Location-Support System. In *6th ACM MOBICOM*, Boston, MA, August 2000.
- [PM07] Evangelos Papadopoulos and Michael Misailidis. On differential drive robot odometry with application to path planning. In *Proceedings of the European Control Conference*, pages 5492–5499, 2007.
- [PMBT01] Nissanka B. Priyantha, Allen K.L. Miu, Hari Balakrishnan, and Seth Teller. The Cricket Compass for context-aware mobile applications. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 1–14, New York, NY, USA, 2001. ACM.
- [Pow78] M.J.D Powell. A fast algorithm for nonlinearly constrained optimization calculations. *Springer Verlag*, 630, 1978.
- [PUH⁺07] M. C. Perez, J. Urena, A. Hernandez, C. De Marziani, A. Jimenez, J. M. Villadangos, and F. Alvarez. Ultrasonic beacon-based local positioning system using loosely synchronous codes. In *Proc. IEEE Int. Symp. Intelligent Signal Processing WISP 2007*, pages 1–6, 2007.
- [Rub09] Mari Carmen Perez Rubio. *Generación y Correlación eficiente de códigos binarios derivados de conjuntos de secuencias complementarias para sistemas ultrasónicos*. PhD thesis, Universidad de Alcalá de Henares, 2009.
- [SD97] X.-M. Shen and L. Deng. Game theory approach to discrete H-infin; filter design. 45(4):1092–1095, 1997.

- [SFMdSPC01] W. Soares-Filho, J. Manoel de Seixas, and L. Pereira Caloba. Principal Component Analysis for classifying passive sonar signals. In *Proc. IEEE Int. Symp. Circuits and Systems ISCAS 2001*, volume 3, pages 592–595, 2001.
- [SG05] P. Steggles and S. Gschwind. The Ubisense smart space platform. In *Proceedings of the Third International Conference on Pervasive Computing*, volume 191, 2005.
- [Sim03] S. N. Simic. A learning-theory approach to sensor networks. *IEEE Pervasive Computing*, 2(4):44–49, 2003.
- [Sim05] D. Simon. To game theory approach to constrained minimax state estimation. Technical report, Department of Electrical Engineering, Cleveland State University, 2005.
- [SJP⁺09] F. Seco, A. R. Jimenez, C. Prieto, J. Roa, and K. Koutsou. A survey of mathematical methods for indoor localization. In *Proc. IEEE Int. Symp. Intelligent Signal Processing WISP 2009*, pages 9–14, 2009.
- [Som58] Duncan M'Laren Young Sommerville. *An introduction to the geometry of n dimensions*. Dover Publications, 1958.
- [SRB01] C. Savarese, J. M. Rabaey, and J. Beutel. Location in distributed ad-hoc Wireless Sensor Networks. In *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP '01)*, volume 4, pages 2037–2040, 2001.
- [SS02] Dong-Ho Shin and Tae-Kyung Sung. Comparisons of error characteristics between TOA and TDOA positioning. 38(1):307–311, 2002.
- [ST92] U. Shaked and Y. Theodor. H-infin;-optimal estimation: a tutorial. In *Proc. 31st IEEE Conf. Decision and Control*, pages 2278–2286, 1992.
- [TK10] Michael R. Thomas K., Lothar M. Evaluating the behaviour of museum visitors using RFID. In *Proc. Indoor Positioning & Indoor Navigation (IPIN)*, 2010.
- [TORC03] F. Thomas, E. Ottaviano, L. Ros, and M. Ceccarelli. Coordinate-free formulation of a 3-2-1 wire-based tracking device using Cayley-Menger determinants. In *Proc. IEEE Int. Conf. Robotics and Automation ICRA '03*, volume 1, pages 355–361, 2003.
- [TR05] F. Thomas and L. Ros. Revisiting trilateration for robot localization. *IEEE Transactions on Robotics*, 21(1):93–101, 2005.

- [Usp48] J. V. Uspensky. *Theory of Equations*. McGraw-Hill, 1948.
- [VDH⁺07] A. Varshavsky, E. Delara, J. Hightower, A. Lamarca, and V. Otsason. GSM indoor localization. *Pervasive and Mobile Computing*, 3(6):698–720, December 2007.
- [VLHdL07] Alex Varshavsky, Anthony LaMarca, Jeffrey Hightower, and Eyal de Lara. The SkyLoc floor localization system. In *PERCOM '07: Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications*, pages 125–134, Washington, DC, USA, 2007. IEEE Computer Society.
- [VUM⁺07] J. M. Villadangos, J. Urea, M. Mazo, A. Hernandez, C. De Marziani, M. C. Perez, F. Alvarez, J. J. Garcia, A. Jimenez, and I. Gude. Ultrasonic local positioning system with large covered area. In *Proc. IEEE Int. Symp. Intelligent Signal Processing WISP 2007*, pages 1–6, 2007.
- [WHFG92] Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons. The Active Badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102, 1992.
- [WJH97] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, 1997.
- [Xil08] Xilinx. Virtex-5 FPGA user guide, Mayo 2008.
- [YB96] B. Yamauchi and R. Beer. Spatial learning for navigation in dynamic environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 26(3):496–505, 1996.
- [Yim08] Jaegeol Yim. Introducing a decision tree-based indoor positioning technique. *Expert Syst. Appl.*, 34:1296–1302, February 2008.
- [YTBJ08] Junlin Yan, C. Tiberius, G. Bellusci, and G. Janssen. Feasibility of Gauss-Newton method for indoor positioning. In *Proc. IEEE/ION Position, Location and Navigation Symp*, pages 660–670, 2008.
- [ZH06] Er-Jie Zhong and Ting-Zhu Huang. Geometric dilution of precision in navigation computation. In *Proc. Int Machine Learning and Cybernetics Conf*, pages 4116–4119, 2006.